

Discussion

Finite Element Modelling of a Field-Sensed Magnetic Suspended System for Accurate Proximity Measurement Based on a Sensor Fusion Algorithm with Unscented Kalman Filter

Amor Chowdhury ¹ and Andrej Sarjaš ^{2,*}

¹ Margento R&D, Gosposvetska cesta 84, Maribor 2000, Slovenia; amor.chowdhury@margento.com

² Faculty of Electrical Engineering and Computer Science, University of Maribor, Smetanova 17, Maribor 2000, Slovenia

* Correspondence: andrej.sarjas@um.si; Tel.: +386-2220-7413

Academic Editor: Andreas Hütten

Received: 7 July 2016; Accepted: 7 September 2016; Published: 15 September 2016

Abstract: The presented paper describes accurate distance measurement for a field-sensed magnetic suspension system. The proximity measurement is based on a Hall effect sensor. The proximity sensor is installed directly on the lower surface of the electro-magnet, which means that it is very sensitive to external magnetic influences and disturbances. External disturbances interfere with the information signal and reduce the usability and reliability of the proximity measurements and, consequently, the whole application operation. A sensor fusion algorithm is deployed for the aforementioned reasons. The sensor fusion algorithm is based on the Unscented Kalman Filter, where a nonlinear dynamic model was derived with the Finite Element Modelling approach. The advantage of such modelling is a more accurate dynamic model parameter estimation, especially in the case when the real structure, materials and dimensions of the real-time application are known. The novelty of the paper is the design of a compact electro-magnetic actuator with a built-in low cost proximity sensor for accurate proximity measurement of the magnetic object. The paper successively presents a modelling procedure with the finite element method, design and parameter settings of a sensor fusion algorithm with Unscented Kalman Filter and, finally, the implementation procedure and results of real-time operation.

Keywords: accurate proximity measurement; sensor fusion algorithm; Unscented Kalman Filter; finite element modelling

1. Introduction

Linear proximity sensors (LSPs) with mid- and low-range measurement capabilities are devices that are used widely in many industrial and non-industrial applications. They are mostly used to determine the displacement, direction of movement, orientation, speed, etc., of a measured object. The LSP exploits different physical principles of operation, where capacitive, inductive, ultrasonic, optical and magnetic phenomena are the most commonly deployed in a sensing operation. Many of these physical phenomena, especially accurate optical technology and the ultrasonic principle, require complex pre- and post-processing operations which, unintentionally, result in a high price and relatively large dimensions of the measuring unit. The dimensional obstacles and the high price of the sensor devices often prevent installation of precise sensing technology on small/miniature and low cost devices. In the time of high expansion and pervasive sensing technologies, especially in the field of miniature sensors, as well as a highly efficient processing unit of relatively small dimensions and price, they offer many applicable solutions which can effectively replace many complex and expensive

solutions. In the last two decades devices using magnetic phenomena based on the Hall effect (HE) have been used widely in many applications due to their low cost, small dimensions, and simple, reliable and effective operation [1]. Application solutions with HE sensors have been demonstrated in many different industrial areas such as: automotive, aerospace, aviation industries etc., and in many of the vastly different engineering fields like electronics, construction, mechanical, medical and computer engineering [1,2]. HE is a well-known technique for the measurement and detection of rotor orientation and positioning in electromechanical machines [3,4], contactless current sensing [2], linear displacement in electromagnetic linear actuators, as well as shaft angle measurement on mechatronic haptic interfaces [5,6].

This paper deals with a relatively small electromagnetic actuator with an integrated proximity sensor (EMAwS). The proximity sensor is based on the measurement of magnetic field density with a ratiometric Hall effect sensor. The electromagnetic actuator is composed of an electromagnet with a HE sensor mounted on the edge surface, perpendicular to the electromagnet flux linkage. The second part of the system is an actuated body with an attached permanent magnet (PM), placed near the electromagnet surface with the HE sensor. The permanent magnet was aligned parallel to the HE sensor. The actuated body/piston can be contactless actuated vertically or horizontally. Such a EMAwS system with regard to the physical construction is widely known as a levitation/suspension [7,8] or a contactless horizontal positioning system [2]. The basic principle of operation is the production of an electromagnetic force on the body, driven by an electric current. Electric current can be driven bidirectionally or unidirectionally in regard to the control scheme and operation principle. The main advantage of such a system is no mechanical contact, low friction, high efficiency and mechanical reliability. According to the previously listed advantages, such a system can be used as a precise positioning system in different industrial applications. There are many different applications of levitation/suspension systems such as; high-speed transportation systems—maglev trains, vibration isolation systems, magnetic bearings, conveyor systems, wind turbines, medical treatment, precision engineering industry, etc. The magnetic levitation and positioning is known as a highly nonlinear, complex and unstable dynamic system, where the proper control algorithm and precise data acquisition with the sensor system play an important role in the device's operation. Many of the studies and published papers have dealt with the design and analysis of the feedback controller. There are many presented solutions based on linear and nonlinear control configurations, especially the simple Proportional Integral Derivative or Proportional Derivative (PID/PD) controller structure presented by Li and Lin [8,9], the robust \mathcal{H}_2 , \mathcal{H}_∞ controller paradigm used by Li [7] and more comprehensive approaches, with linear and nonlinear approaches based on Linear-Quadratic-Gaussian (LQG) controller, sliding mode, back stepping and the feedback linearization design technique in combination with intelligent fuzzy and neural systems presented by the authors Mehrtash, Shameli, Elbuken, Lin, Yang, Gentili and Kashif [10–17].

In most research about levitation and magnetic positioning there is a profound lack of details about position measurement, especially in the application with HE sensors. For this reason, the main focus of this paper is the research and development of an accurate distance measurement with a HE sensor in the magnetic positioning system. The accurate sensing of the suspended object is key, crucial for the further analysis of the control algorithm, particularly where positioning of the suspended object is required at very long range. The most challenging subject of the sensor application was the suppression of the substantial external influences, measurement noise, and other unknown time varying uncertainties of the HE element. The external influences are described primarily as influences of the electromagnet and presence of the PM magnetic fields to the HE sensor. Both quantities have a highly nonlinear connection. Nonlinearity depends on the system dimensions, distance between the object and electromagnet, the drive coil current and other parameters, such as variable resistance and inductance due to electromagnet heating and other unknown factors. With the aim to suppress the aforementioned disturbances and to mitigate unknown influences on the position measurement, the Sensor Fusion Algorithm (SFA) was deployed. The SFA was based on the redundant classification,

where two sources provide information of the positioning measurement [18]. The first source of the SFA was direct measurement from the HE sensor and the second was the state estimation from the mathematical model. The Unscented Kalman Filter—UKF was employed as a nonlinear state estimation technique [19,20]. The Kalman filter—KF is a state correction, bias and noise suppression algorithm widely used in many applications [21]. The crucial starting point for proper operation of the KF algorithm is accurate model dynamics and the initial settings of the error covariance matrices. The UKF algorithm is intended for nonlinear estimation and has better features, like the other linear-based estimation KF algorithm. The presented paper deals with the derivation of the eligible model for accurate and reliable state estimation. There were many studies with model derivations which relied solely on an analytical approach based on physical laws, presented by Naumović and Hajjaji [22,23]. Some of the authors used the linear model structure Autoregressive exogenous—ARX and different identification technique in frequency domain, presented by Yemeo and Shameli [24,25]. Mostly such approaches involved more or less simplification and approximation which, consequently, caused imprecise model dynamics. On the other hand, it is also important to note that the derived models in many studies are used further for the controller design [24–26]. The main task of the controller is ensuring the stability and robustness of the feedback system with regard to the plant uncertainty and reference changes [7–9]. It is also evident that many authors used the same mathematical model for different configurations of the levitation system; levitating PM or metallic object. In this case, the tracking performance for a wider range of levitation deteriorates drastically, because the deviation from the nominal operating point increases highly. This issue, i.e., how important the accuracy of the plant model is, was discussed by Hajjaji [23]. The levitating magnet or metal object has a different relation between distance and coil driven current on the electromagnetic force, which is applied to the levitating body. Especially, the levitating PM has a very complex analytical description of the magnetic force, which depends on the estimation of the magnetic moments. To avoid complexity of the analytical model the Finite Element Method (FEM) was used for electromechanical systems. Magnetic field modelling is an important research area, where the real quantities are the effects of the magnetic field such as; force, electromotive force—EMF, inductance, etc. [27]. The FEM technique is used often for structure design, modelling and efficiency measurement of different kinds of electromagnetic machines [27,28]. In the present work, the FEM technique was used for accurate modelling of the real system, with known dimensions and material characteristics. The aim of the FEM modelling was to acquire the relation between different quantities, such as driven coil current, magnet proximity, coil dimension etc. to the dynamics of the electromagnetic force. The derived model has key importance on the reliability of the used sensor fusion algorithm with UKF which, consequently, affects the accuracy of the proximity measurement. The efficiency of the FEM modelling approach for proximity measurement with UKF, will be presented and tested on the real levitation system with PM.

The study is organised as follows: problem formulation and the EMaWS structure is described in Section 2. The modelling procedure with FEM and parameter optimization is presented in Section 3. The design of the data fusion algorithm with UKF is discussed in Section 4. Validation and the effectiveness of the measurement system are presented in Section 5. Finally, the paper is concluded in Section 6.

2. Experimental Prototype and Modelling of the Electromagnetic Actuator with Integrated Proximity Sensor

The experimental prototype of the EMaWS electromagnetic actuator with integrated proximity HE sensor is composed from three main parts: Electromagnet, ratiometric Hall Effect sensor and actuated body with PM. The ratiometric HE sensor is attached on the electromagnet close to the side of the PM body, centre aligned with the exit flux linkage field lines. The actuated PM body is positioned along the electromagnet, parallel to the HE sensor. The body can be moved vertically or horizontally, depending on the actuator structure. Two different structure types of EMaWS are depicted in Figure 1.

Figure 1 shows electromagnet—1 with driven current i , linear Hall Effect sensor—2 and actuated body with permanent magnet—3. Variable d is the distance between the actuated body and the electromagnet. The following research will not be relayed directly on the specified structure (a) or (b) of the EMAwS depicted in Figure 1, but on the modelling of the magnetic force F_e for the proximity measurement d . The modelling of the magnetic force applied to the magnetic body is principally the same for systems (a) and (b) in Figure 1. For the electromagnetic force modelling and for further data fusion algorithm the state space description of the system is taken.

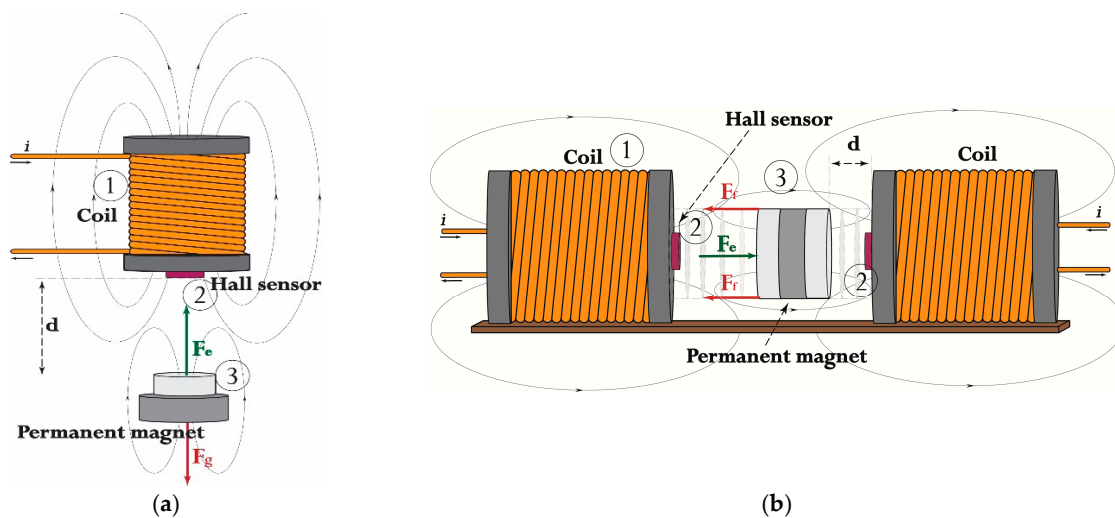


Figure 1. Two electromagnetic actuators with integrated proximity HE sensor-EMAwS; (a) Vertical movement of the body and (b) Horizontal movement of the body.

The state space representation of the system is:

$$y^{(n)}(t) = g(x(t)) + f(x(t))u(t) \quad (1)$$

where the terms $y(t) = [y_1(t), y_2(t), \dots, y_m(t)] \in \mathbb{R}^m$, $x(t) = [x_1(t), x_2(t), \dots, x_m(t)] \in \mathbb{R}^{mn}$ and $u(t) = [u_1(t), u_2(t), \dots, u_m(t)] \in \mathbb{R}^m$ denote the system output vector, states vector and system input vector, respectively. The functions $g(t) \in \mathbb{R}^m$ and $f(t) \in \mathbb{R}^m$ represent smooth nonlinear uncertainty functions and they are assumed to be bounded. With regard to the law of motion the model structure is:

$$m \frac{d^2 x(t)}{dt^2} = \pm F_{g/f}(t) \mp F_e(x, i, t) \quad (2)$$

and the coil dynamic equation is:

$$U(t) = R \cdot i(t) + L \frac{di(t)}{dt} + km(x) \frac{dx(t)}{dt} \quad (3)$$

where $U(t)$, $i(t)$, R , L , $k_m(x)$, x , m , $F_{fg}(t)$, $F_e(x, i, t)$ are the applied voltage, coil current, winding resistance, winding inductance, induction constant, distance, body mass, opposite force and magnetic force, respectively. The meaning of opposite force depends on the system structure in Figure 1 and can be treated as gravity force $F_g = mg$ for system (a) and the friction force $F_f = kh$ for the system (b). The parameter k is the friction coefficient. Calculation of the magnetic force between two magnets is a complex task and can be done with the Gilbert and Ampere model [29]. With regard to the mentioned models the magnetic force F_e between the actuated body and the electromagnet can be determined with inspection of the magnetic field as a function of the separation distance presented by Naumović and Hajjaji [22,23,29,30]. It can be done with the integral method, where the solenoid is modelled as a volume current density and the permanent magnet as a surface current density

around its circumference presented by Furlani and Robertson [31,32]. The derived magnetic force models are mostly highly complex functions, which can be used only for offline analysis, parameter estimation and structure optimization [32]. Such models have very limited use in real-time systems, especially in the systems with fast execution demand and the systems with modest computational power. In the presented research the magnetic force model will be used in the UKF algorithm in the systems with relatively short execution time demand and feedback controller. Algorithms, UKF and feedback controller need to be executed in each sampling period iteratively. The main focus of the modelling is obtaining an accurate model dynamic with an adequately simple structure.

3. Magnetic Force Modelling with the Finite Element Method

Partial nonlinear equations arise in mathematical modelling in many diverse areas, such as material science, fluid dynamics, electromagnetism, economics, etc. [33]. In most cases the equations of the described system are so complicated that their solution in purely analytical form is impossible or impractical. Due to the complexity, many times the circumstances have compelled users to search for approximate solutions to the unknown analytical solutions. The FEM is one of the numerical techniques used for finding approximate solutions of partial differential equations with known boundary values [34]. In the presented paper the FEM technique is used for modelling and analysis of the electromagnetic force F_e and EMAwS system. The open source 2D Finite Element Method Magnetics (FEMM) created by Meeker [35] was used. FEMM offers broad coupling with different external simulation and analysis software and self-created scripts in LUA-language. Such external coupling and self-created scripts' capability allows many options in simulation, analysis and measurement. Each created model's parameters, dimensions, electrical circuit properties, etc. in FEMM can be controlled by coupled external software or the self-created scripts in LUA. We used the coupling possibility with MATLAB software, where MATLAB scripts are controlling the multiphysics model in FEMM, similar to the work presented by Benamimour [36].

The analysis of the electromagnetic force F_e was started by drawing the EMAwS system in FEMM. The model drawing is an important task, where all material characteristics, geometric and circuit parameters, measurement units, mesh polygonal angle (grid generation) are determined. The system (a) in Figure 1 will be examined for further analysis. The system parameters are recorded in Table 1 and in Figure 2.

Table 1. EMAwS system parameters.

Parameters	Value
Solenoid high (h_s)	25 mm
Solenoid flange high (h_{se})	5 mm
Solenoid flange diameter (w_s)	40 mm
Centre hole diameter (p)	5 mm
Inside winding diameter (w_{sin})	20 mm
Ferrite magnetic permeability (μ/μ_0)	450
Permanent magnet-neodymium	N52
Permanent magnet diameter (w_m)	11 mm
Permanent magnet high (h_m)	5 mm
Coil wire diameter AGW30	0.255 mm
Number of turns	30
Weight of the magnet	4.35 g

All the materials, wire characteristic, number of coil turns, driven current, PM characteristic were assigned in the FEMM software as depicted in Figure 3. The picture in Figure 3 represents a 2D-EMAwS model in central cross-section view, with computational boundary.

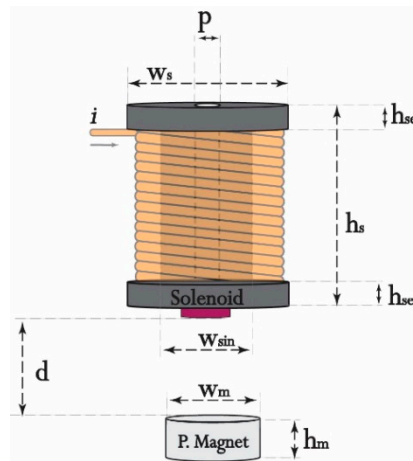


Figure 2. The EMAwS characteristics.

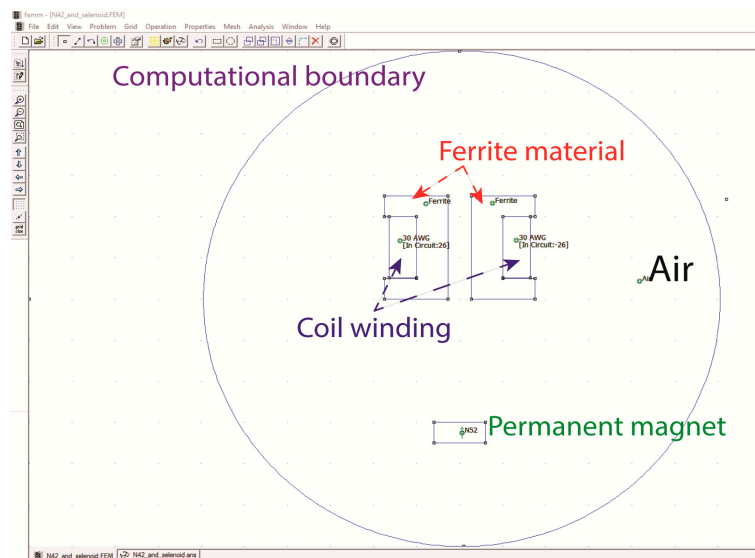


Figure 3. Plotted EMAwS in FEMM software.

In Figure 3, with the selection of solver precision ($1e^{-8}$) and minimal angle value setting (30) in a millimeter scale for the inside computational boundary surface, we get a computational mesh with 17,257 nodes. The selection of the solver parameters (solver precision and mesh angle) is crucial in the accuracy of the simulation results. In our approach the solver parameters were selected experimentally in a manner to acquire accurate and reliable results in a reasonable time. By increasing the solver precision and angle value of the FEM procedure, the computational effort was greatly increased, wherein the accuracy of the simulation was insignificant compared to the computation time used. In regard to the drawn model and solver parameters' selection, the simulation lasted approximately 4 min on a Windows 10-based PC, with a i7-3770@3.4GHz CPU and 8GB-RAM memory. The computational mesh in the selected computational boundary surface and calculated quantities with FEM are depicted in Figure 4.

The FEMM software offers calculation of many different quantities and values of the designed electromechanical system. The electromagnetic force F_e for the given system in Figure 4 was calculated via a weighted stress tensor, where the force value components were mapped in a 2D plane.

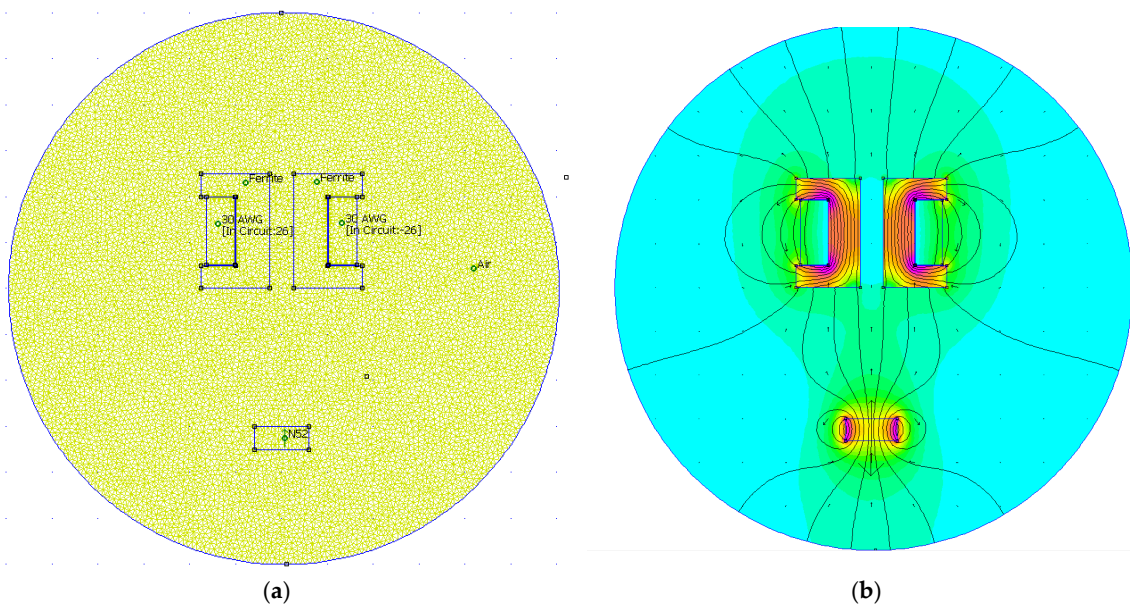


Figure 4. Finite element method with FEMM software for EMAwS. (a) Computational mesh; (b) Calculated values of magnetic force and field density.

3.1. Coupling FEMM with Matlab and Data Analysis

The coupling procedure is attained after designing a model in FEMM software with all the material characteristics. There were many choices for how to achieve and how to handle the objects inside the FEMM from outer programmes or self-defined scripts. As we mentioned before, the Matlab software was used to handle model objects externally in FEMM. Matlab-FEMM coupling has many possibilities for managing model objects and FEMM properties. The whole design of the model and simulation procedure inside the Matlab software was presented by Benamimour [36]. In our case, we used the possibility of simulation commissioning and model objects handling in an already existing FEMM model. The coupled Matlab software was used to control and acquire simulated data from FEMM. Matlab software scheduled the simulation parameters, where the driven coil current and vertical position of the PM were adjusted. The whole experiment was based on the measurement of the electromagnetic force, which was acquired from the FEMM at the exact specified value of the current and PM position. The current value was changed on interval $[0 \text{ A} \div 1.4 \text{ A}]$ by steps of 0.2 A, where position was changed on interval $[15 \text{ mm} \div 45 \text{ mm}]$ by steps of 0.5 mm. The simulation schedule is depicted in Figure 5.

The simulated values are further used for model parameters estimation given in Equation (2). The obtained results from the simulation are presented in Figures 6 and 7.

Figures 6 and 7 present the nonlinear dependence of the electromagnetic force between coil current and the PM position.

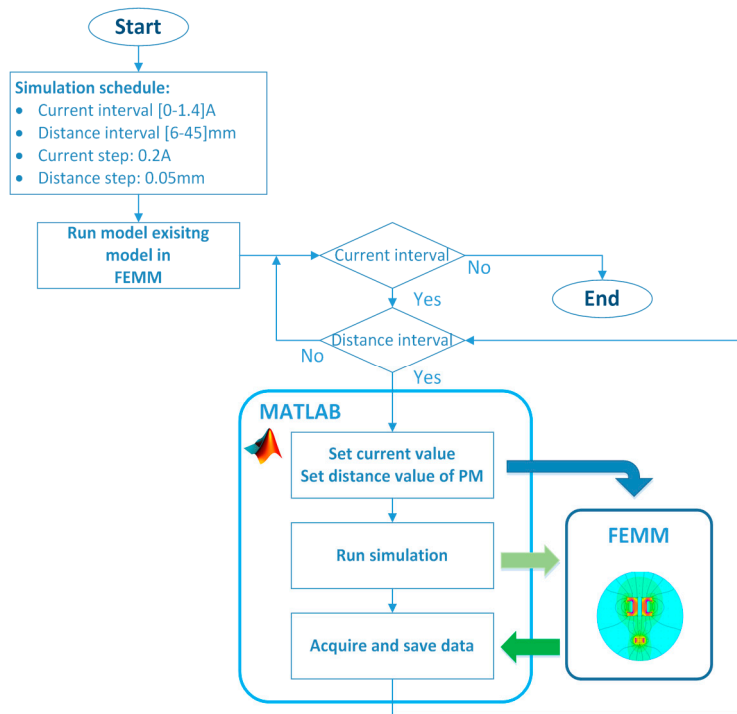


Figure 5. Simulation schedule with coupled Matlab and FEMM software.

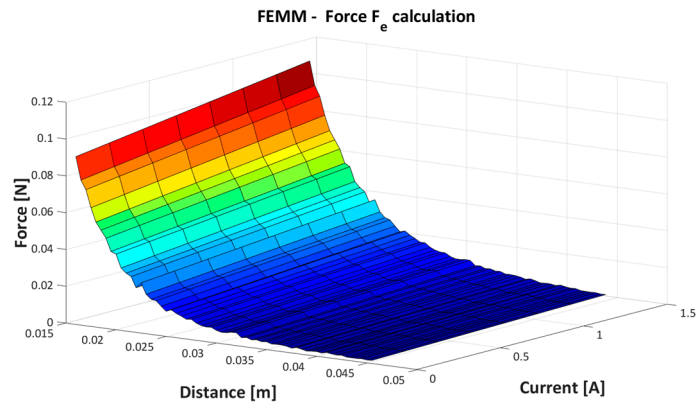


Figure 6. Simulation results of the FEMM calculation.

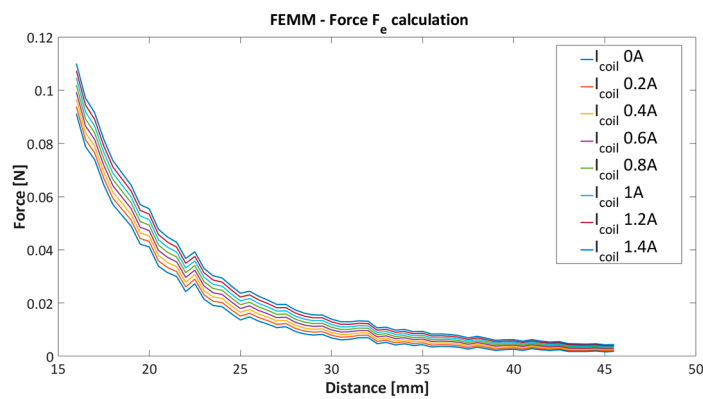


Figure 7. FEMM force calculation for different coil current.

3.2. Modelling of Electromagnetic Force F_e

The modelling of the electromagnetic force will be presented from the obtained simulation results from Matlab and FEMM software. The modelling of the electromagnetic F_e has been based on the function fitting procedure to the set of the simulated result data's with preselected model structure. Model estimation and data fitting was performed with the quadratic programming (QP) method. The QP technique is a well-known and efficient approach for convex optimization with introduced, boundaries, inequalities and equalities [37]. Before the objective function for optimization will be derived, the same optimization relaxation will be assigned. The modelling of the F_e will be considered only on the system with fixed structure and geometrical form. The current and distance dependence by given fixed preselected system will be considered. From the simulated data presented in Figure 7 it is straightforward to recognise that the current dependence is completely linear to the force F_e . This assumption can be proven with the calculated difference between each adjacent current's characteristic in Figure 7. Each current characteristic has been labelled as I_k , where k represents corresponding current value from 0 A to 1.4 A by steps of 0.2 A. The linearity test L_p of current to the F_e is given with:

$$L_p = I_k - I_{k-1} \quad (4)$$

where p represents the number of pairs of adjacent difference characteristics. In the presented case with 8 current characteristics $k \in [0 \div 8]$ in Figure 7, we have seven difference pairs; $p = 7$. With deviation and mean calculation of the L_p we have proved the linear dependence of the current to the force F_e . The calculated values are presented in Table 2.

Table 2. Calculated difference characteristics L_p with mean and standard deviation values.

p	L_d	Current Characteristic Difference	Mean Value	Standard Deviation
1	L_1	[0.2–0] A	0.0012	0.6740
2	L_2	[0.4–0.2] A	0.0012	0.6740
3	L_3	[0.6–0.4] A	0.0012	0.6740
4	L_4	[0.8–0.6] A	0.0012	0.6740
5	L_5	[1–0.8] A	0.0012	0.6740
6	L_6	[1.2–1] A	0.0012	0.6740
7	L_7	[1.4–1.2] A	0.0012	0.6740

The distance dependence needs to be determined in regard to the linear correlation of current and force. The distance and force dependence was estimated with a curve fitting algorithm and the QP optimization technique. The QP optimization technique allows finding an optimal solution in the given interval of search parameters with selected equality and inequality constraints. In the given case, only positive solutions are allowed. The optimization procedure was divided into two stages. In the first stage the general model structure of the model was specified, where the polynomial parameters stay uncertain. The corresponding preselected model was fitted for each current characteristic in Figure 7. After the first stage of the optimization, we get a set of different models with fixed structure and different polynomial coefficients. In regard to Figure 7 and the simulation schedule we derived eight different models. The second stage was intended to estimate polynomial coefficients, where each coefficient in the polynomial has eight values from a fitted model. Each coefficient is also approximated with a new coefficient polynomial function. The coefficient polynomial functions are inserted further in the preselected model structure from the previous stage one.

3.3. Model Structure Selection F_e —Stage One

The first stage begins with selection of the simple model structure. The best two model candidates were selected after a few iterations and with the assumption of the current-force linearity. The best model candidates for F_e are:

$$F_{1e}(d) = ad^{-4} + bd^{-2} \tag{5}$$

$$F_{2e}(d) = cd^{-4} \tag{6}$$

where coefficients a, b, c are estimated with QP optimization, for each separated current characteristic. The variable d is PM proximity. Selected residual functions for QP optimization are:

$$J_1(d) = \sum_{i=1}^n (y_i - F_{1e}(d))^2 = \sum_{i=1}^n \left(y_i - (ad^{-4} + bd^{-2}) \right)^2, \tag{7}$$

$$J_2(d) = \sum_{i=1}^n (y_i - F_{2e}(d))^2 = \sum_{i=1}^n \left(y_i - (cd^{-4}) \right)^2, \tag{8}$$

where n is a number of simulation data from FEMM. The parameters a, b, c are solution of the QP programming technique with residual Equations (7) and (8). The QP solutions with corresponding matrixes are:

$$\begin{aligned} &\min \frac{1}{2} \mathbf{x}^T \mathbf{P} \mathbf{x} + \mathbf{N} \mathbf{x} \\ &s.t. \begin{cases} \mathbf{A}_{eq} \mathbf{x} = \mathbf{B}_{eq} \\ \mathbf{l}_b \leq \mathbf{x} \leq \mathbf{u}_b \end{cases} \end{aligned} \tag{9}$$

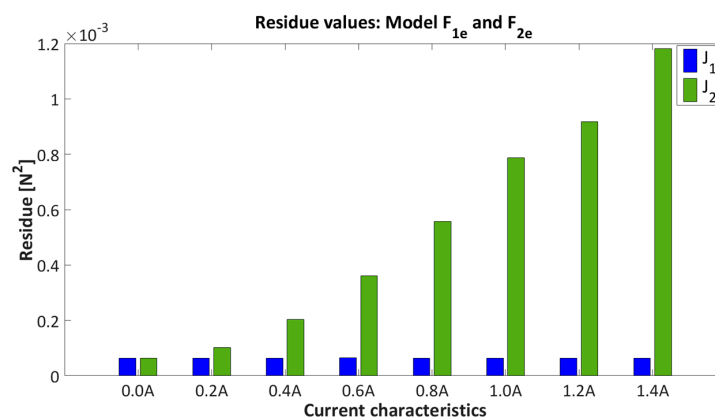
$$\begin{aligned} \mathbf{P}_{F_{e1}} &= \begin{bmatrix} 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & 1 & 0 & \dots & 0 \\ \vdots & \vdots & 0 & 1 & \dots & 0 \\ \vdots & \vdots & & & \ddots & \\ 0 & 0 & 0 & 0 & \dots & 1 \end{bmatrix}_{n+2 \times n+2}, \quad \mathbf{N}_{F_{e1}} = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ 0 \end{bmatrix}_{n+2 \times 1}, \quad \mathbf{A}_{eq \ F_{e1}} = \begin{bmatrix} d_1^{-4} & d_1^{-2} & 1 & 0 & \dots & 0 \\ d_2^{-4} & d_2^{-2} & 0 & 1 & \dots & 0 \\ d_3^{-4} & d_3^{-2} & 0 & 0 & \dots & 0 \\ \vdots & \vdots & & & \ddots & \\ d_n^{-4} & d_n^{-2} & 0 & 0 & \dots & 1 \end{bmatrix}_{n \times n+2}, \\ \mathbf{B}_{eq \ F_{e1}} &= \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}_{n \times 1}, \quad \mathbf{l}_{b_{F_{e1}}} = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ \vdots \\ 0 \end{bmatrix}_{n+2 \times 1}, \quad \mathbf{u}_{b_{F_{e1}}} = \begin{bmatrix} 10^{-7} \\ \vdots \\ \vdots \\ \vdots \\ 10^{-7} \end{bmatrix}_{n+2 \times 1}, \quad \mathbf{x} = \begin{bmatrix} a \\ b \\ 0 \\ \vdots \\ 0 \end{bmatrix}_{n+2 \times 1}, \\ \mathbf{P}_{F_{e2}} &= \begin{bmatrix} 0 & 0 & 0 & 0 & \dots & 0 \\ \vdots & 1 & 0 & 0 & \dots & 0 \\ \vdots & 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & & & \ddots & \\ 0 & 0 & 0 & 0 & \dots & 1 \end{bmatrix}_{n+1 \times n+1}, \quad \mathbf{N}_{F_{e2}} = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ \vdots \\ 0 \end{bmatrix}_{n+1 \times 1}, \quad \mathbf{A}_{eq \ F_{e2}} = \begin{bmatrix} d_1^{-4} & 1 & 0 & \dots & 0 \\ d_2^{-4} & 0 & 1 & \dots & 0 \\ d_3^{-4} & 0 & 0 & \dots & 0 \\ \vdots & \vdots & & \ddots & \\ d_n^{-4} & 0 & 0 & \dots & 1 \end{bmatrix}_{n \times n+1}, \\ \mathbf{B}_{eq \ F_{e2}} &= \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}_{n \times 1}, \quad \mathbf{l}_{b_{F_{e2}}} = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ \vdots \\ 0 \end{bmatrix}_{n+1 \times 1}, \quad \mathbf{u}_{b_{F_{e2}}} = \begin{bmatrix} 10^{-7} \\ \vdots \\ \vdots \\ \vdots \\ 10^{-7} \end{bmatrix}_{n+1 \times 1}, \quad \mathbf{x} = \begin{bmatrix} c \\ 0 \\ \vdots \\ 0 \end{bmatrix}_{n+1 \times 1} \end{aligned} \tag{10}$$

Results of model fitting F_{1e} and F_{2e} over current characteristic from 0 A to 1.4 A, are presented in Table 3 and Figures 8 and 9.

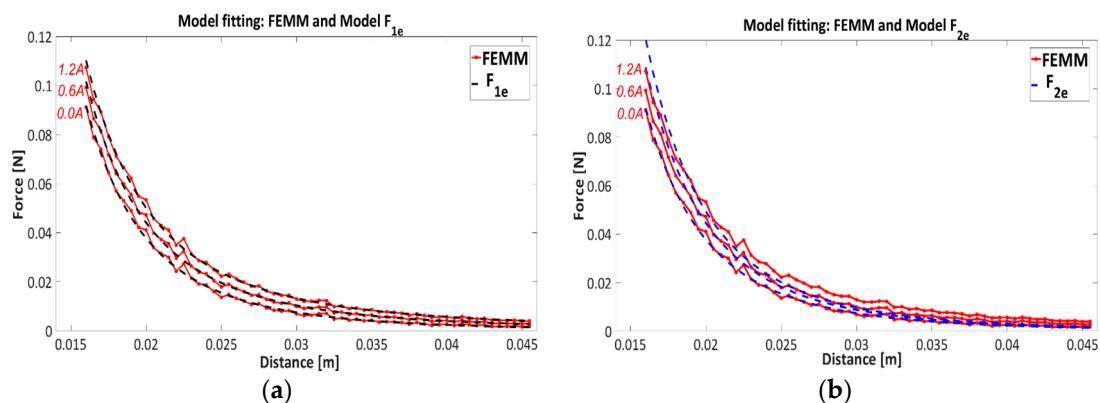
Table 3. Parameter estimation of the model F_{1e} and F_{2e} , by current characteristics 0 A–1.4 A.

Current	Model F_{1e}			Model F_{2e}	
	a	b	Residual (J_1)	c	Residual (J_2)
0 A	5.999×10^{-9}	7.587×10^{-8}	6.409×10^{-5}	6.031×10^{-9}	6.424×10^{-5}
0.2 A	5.98×10^{-9}	88.6×10^{-8}	6.421×10^{-5}	6.415×10^{-9}	1.13×10^{-4}
0.4 A	6.012×10^{-9}	174.4×10^{-8}	6.399×10^{-5}	6.781×10^{-9}	2.033×10^{-4}
0.6 A	6.005×10^{-9}	263.4×10^{-8}	6.437×10^{-5}	7.135×10^{-9}	3.615×10^{-4}
0.8 A	5.985×10^{-9}	357.1×10^{-8}	6.427×10^{-5}	7464×10^{-9}	5.512×10^{-4}
1 A	5.994×10^{-9}	450.6×10^{-8}	6.395×10^{-5}	7.775×10^{-9}	7.867×10^{-4}
1.2 A	5.989×10^{-9}	543.1×10^{-8}	6.397×10^{-5}	7.892×10^{-9}	9.183×10^{-4}
1.4 A	6.007×10^{-9}	637.2×10^{-8}	6.403×10^{-5}	8.134×10^{-9}	12.437×10^{-4}

Figure 8 represents the residue values for model estimation F_{1e} and F_{2e} .

**Figure 8.** Residue value of model fitting F_{1e} , F_{2e} over current characteristics from 0 A to 1.4 A.

The model fitting on current characteristics is presented in Figure 9, where Figure 9a,b show the fitting properties with models F_{1e} and F_{2e} , respectively. Only three current characteristics 0 A, 0.6 A and 1.2 A are presented for better clarity of the results. It is obvious from Table 3, Figures 8 and 9, that model F_{1e} has better data matching to the current characteristics than model F_{2e} . From Table 3 it can be seen that the residual values indicate that model structure F_{1e} has quite accurate fitting properties over the whole area, where residual values remain constant with neglect deviation in regard to the residual values of the model F_{2e} . The model F_{1e} will be considered for further analysis.

**Figure 9.** Comparison of model fitting with models F_{1e} —(a) and F_{2e} —(b) on current characteristics 0 A, 0.6 A and 1.2 A.

3.4. The F_{1e} Coefficient Estimation and F_e Model Derivation—Stage Two

After model structure derivation F_{1e} the next step of the optimization procedure is estimation of coefficient functions. The coefficient function describes the coefficient change in model F_{1e} , where it is apparent from Table 3 that, the estimated model F_{1e} has variable coefficients. Figure 10 shows the parameter change of model F_{1e} with regard to the current characteristics.

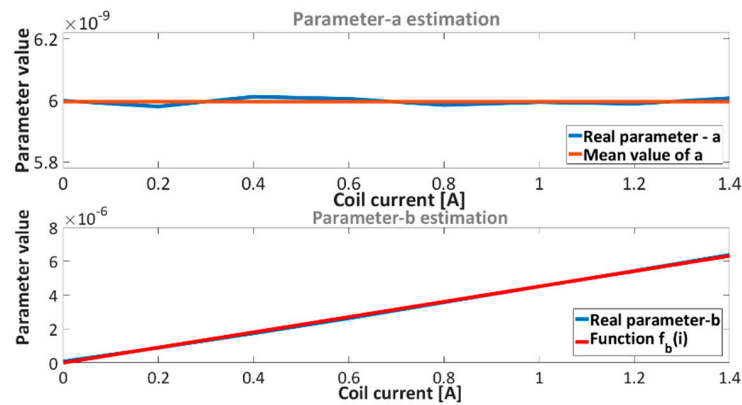


Figure 10. Estimated parameters (top) a and (bottom) b .

With regard to Figure 10 only parameter b needs to be estimated, where parameter a has a constant value with neglected deviation. The mean value of parameter a is 5.996×10^{-9} . Parameter b can be approximate with linear function. Such function indicates linear dependence between force and current, which was confirmed before with Table 2. The selected linear function of parameter b is:

$$f_b(i) = mi \quad (11)$$

where $f_b(i)$ is a b parameter function of current and unknown parameter m . The same optimization procedure was used as for model Equations (5) and (6), where residual function is:

$$J_3 = \sum_{j=1}^n (b_j - f_b(i))^2 = \sum_{j=1}^n (b_j - (mi))^2 \quad (12)$$

The obtained results are presented in Table 4 and in Figure 10.

Table 4. Parameter b estimation.

Parameter a	Parameter $b-(i)$	
Mean \bar{a}	m	Residual (J_3)
5.996×10^{-9}	4.509×10^{-6}	2.0486×10^{-14}

Fitting solutions of the parameters a and b are presented in Figure 10. After derivation of the parameter a , function $f_b(i)$ and model structure in Equation (5), we get an accurate model of the electromagnetic force F_e . The electromagnetic force model F_e is:

$$\begin{aligned} F_e(x, i) &= ax^{-4} + f_b(i)x^{-2}, \\ F_e(x, i) &= 5.996 \times 10^{-9}x^{-4} + 4.509 \times 10^{-6}ix^{-2}. \end{aligned} \quad (13)$$

Table 5 presents the final results of model fitting with model Equations (13), consideration of objective function given in Equation (7) and simulated FEMM results.

The graphical model fitting results with derived model F_e given in Equation (13) and FEMM was already presented in Figure 9a.

Table 5. Residual value of objective function J_1 with final model F_e and FEMM results.

Current	Model $F_e(x, i) = 5.996 \times 10^{-9}x^{-4} + 5.996 \times 10^{-9} \cdot i \cdot x^{-2}$	
	Residual (J_1)	
0 A	6.413×10^{-5}	
0.2 A	6.411×10^{-5}	
0.4 A	6.402×10^{-5}	
0.6 A	6.399×10^{-5}	
0.8 A	6.417×10^{-5}	
1 A	6.399×10^{-5}	
1.2 A	6.401×10^{-5}	
1.4 A	6.416×10^{-5}	

4. Distance Measurement and Sensor Fusion Algorithm

After derivation of a proper mathematical model, the next step was to design an algorithm for accurate position measurement. There are many approaches and estimation algorithms which offer many promising and useful results. The basic problem of distance measurement with a Hall sensor is the noisy and biasing output voltage, which is caused by structural and external factors. In our application, the disturbances on the Hall sensor can be generally defined as: external magnetic field influence, from the driven coil, the temperature dependence and structural imperfections of the Hall element. All influences have a complex nonlinear dependence, where the magnetic field from a driven coil can be prior estimated with static characteristics. With determined static characteristic the disturbances can be mainly suppressed but the accuracy of the measured distance can still remain incorrect due to the other unknown disturbances, voltage drift, sensor noise, imperfect static characteristics etc. The measured static characteristics for the system in Figure 1a with an Allegro MicroSystems A1321 Hall sensor (Allegro MicroSystems, LLC, Worcester, MA, USA) are depicted in Figure 11.

Figure 11a presents the slack linear static characteristic between the coil and Hall voltage, where the Hall voltage and PM distance have a nonlinear relation. Both characteristics in Figure 11 are used for direct proximity measurement with a Hall sensor; conversion from a Hall voltage to PM distance. The characteristics Hall-coil voltage and Hall voltage and PM position are estimated in the same fashion as the characteristic for electrical force F_e given in Equations (7) and (8).

The equations for direct distance measurement with static characteristics compensation are:

$$d_{hall}(v, v_{hall}) = MH(CH(v, v_{hall})), \quad (14)$$

$$CH(v) = v_{hall} - 1.299 \times 10^{-6}v^2 + 9.05 \times 10^{-5}v + 2.544, \quad (15)$$

$$MH(v) = -563.9 \cdot CH(v)^5 + 1351.6 \cdot CH(v)^4 - 1284.2 \cdot CH(v)^3 + 633.1 \cdot CH(v)^2 - 183.9 \cdot CH(v) + 46.9, \quad (16)$$

where $CH(v)$, $MH(v)$, v , v_{hall} , d_{hall} represents the Hall-coil voltage characteristic, magnet-Hall characteristic, coil voltage, Hall voltage and PM proximity, respectively. Direct distance measurement is composed with coil voltage compensation $CH(v)$, after which the PM proximity is obtained from the $MH(v)$ characteristic.

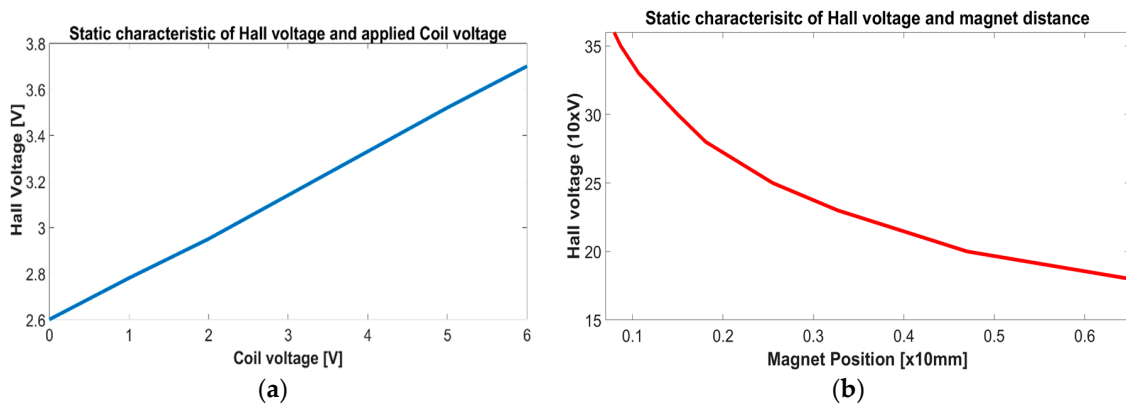


Figure 11. Static characteristics of Hall voltage in regard to applied coil voltage—(a) and the magnet proximity—(b).

4.1. Unscented Kalman Filter and Sensor Fusion Algorithm

The Kalman filter—KF is a broadly used estimation, prediction and sensor fusion algorithm [19,38,39]. There exist many variants of the KF algorithm, where the Extended Kalman Filter—EKF and Unscented Kalman Filter—UKF are particularly used to deal with nonlinear systems. The central operation of the linear KF is the propagation of the Gaussian random variable through the system dynamics, where covariance of the estimation error needs to be minimized. The EKF used the same procedure as the linear KF, where the Gaussian random variable is approximated analytically with Jacobian or Hessian matrices (Taylor series—TS approximation-linearization). Such approximations can, in some cases, introduce large errors in the true posterior mean and covariance of the Gaussian random variable, which may lead to poor performance of the filter [40]. The UKF addresses the issues with approximation of the Gaussian random variables with the Taylor series. Similar to the TS approximation the Unscented Transformation (UT) can be used for forming a Gaussian approximation of the joint distribution of random variables, which are defined in the nonlinear dynamic system. UT transformation deterministically chooses a relatively small amount of the fixed number of sigma points, which capture the true mean and covariance of the Gaussian random state variable [41]. The UT transformation is a method for calculation of the statistics of a random variable which undergoes a nonlinear system. In regard to the TS approximation in the EKF algorithm, the UT transformation is better at capturing the higher order of moments caused by the non-linear transform and is less error prone in regard to the calculation of the Jacobian and Hessian matrices—TS [41–43]. The basic framework of the UKF involves the estimation of the state of the discrete time nonlinear system. The discrete time non-linear system is:

$$\begin{aligned} x_{k+1} &= \mathbf{F}(x_k, u_k, w_k), \\ y_k &= \mathbf{H}(x_k, n_k), \end{aligned} \quad (17)$$

where x_k represents the unobservable states of the system, u_k is a system input and y_k is the measured output signal. The w_k and n_k are process and measurement noise, respectively, with corresponding process noise Q and measurement noise R covariance matrices. The non-linear state \mathbf{F} and output \mathbf{H} functions are known. The UKF algorithm is just a straightforward extension of the UT transformation to the recursive estimation of standard state update equation:

$$\hat{x}_k = (\text{prediction of } x_k) + \kappa_k (y_k - (\text{prediction of } y_k)), \quad (18)$$

where κ_k is Kalman gain and \hat{x}_k is estimated state vector-filter output. The UT transformation sigma points are applied in the new augmented sigma point matrix χ_{k-1}^a , which is generalized obtained from a value of the vector \hat{x}_{k-1} and the state covariance matrix \mathbf{P}_{k-1} [41]. The UKF algorithm equations and calculation procedure are given below.

- Initialization of the UKF- filter:

$$\begin{aligned}\hat{x}_0 &= E[x_0], \\ P_0 &= E[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T].\end{aligned}\quad (19)$$

Variable \hat{x}_0 represents initial states of nonlinear system given in Equation (17) and P_0 is an initial covariance matrix of the state variable x_0 .

- Prediction:

UT-transformation, calculation of $2L + 1$ sigma points:

$$\begin{aligned}\chi_{0,k-1}^a &= \hat{x}_{k-1}, \\ \chi_{i,k-1}^a &= \hat{x}_{k-1} + \sqrt{(L + \lambda) P_{k-1}}, \quad i = 1, \dots, L, \\ \chi_{i,k-1}^a &= \hat{x}_{k-1} - \sqrt{(L + \lambda) P_{k-1}}, \quad i = L + 1, \dots, 2L,\end{aligned}\quad (20)$$

and associated weights:

$$\begin{aligned}\mathbf{W}_0^m &= \lambda / (L + \lambda), \\ \mathbf{W}_0^c &= \lambda / (L + \lambda) + (1 - \alpha^2 + \beta), \\ \mathbf{W}_i^m &= \mathbf{W}_i^c = 1 / (2L + 2\lambda), \quad i = 1, \dots, 2L.\end{aligned}\quad (21)$$

The variable L is the number of system states and λ is the scaling parameter; $\lambda = \alpha^2 (L + k_i) - L$. Parameter α determinates the spreads of the sigma points, k_i is a secondary scaling parameter and β is used to incorporate prior knowledge of the distribution χ_k^a . The weights W^m and W^c represent mean weighting factor and estimation error covariance weighting factor respectively [42].

- Time Update

$$\chi_{k|k-1}^a = \mathbf{F}[\chi_{k-1}^a, u_{k-1}], \quad (22)$$

$$\hat{x}_{k|k-1} = \sum_{i=0}^{2L} \mathbf{W}_i^m \chi_{i,k|k-1}^a, \quad (23)$$

$$P_{k|k-1} = \sum_{i=0}^{2L} \mathbf{W}_i^c [\chi_{i,k|k-1}^a - \hat{x}_{k|k-1}] [\chi_{i,k|k-1}^a - \hat{x}_{k|k-1}]^T + Q_k, \quad (24)$$

$$\kappa_{k|k-1} = \mathbf{H}[\chi_{k|k-1}^a], \quad (25)$$

$$\hat{y}_{k|k-1} = \sum_{i=0}^{2L} \mathbf{W}_i^m \Upsilon_{i,k|k-1}. \quad (26)$$

- Measurement Update Equation

$$P_{y_k y_k} = \sum_{i=0}^{2L} \mathbf{W}_i^c [\Upsilon_{i,k|k-1} - \hat{y}_{k|k-1}] [\Upsilon_{i,k|k-1} - \hat{y}_{k|k-1}]^T + R_k, \quad (27)$$

$$P_{x_k y_k} = \sum_{i=0}^{2L} \mathbf{W}_i^c [\chi_{i,k|k-1}^a - \hat{x}_{k|k-1}] [\Upsilon_{i,k|k-1} - \hat{y}_{k|k-1}]^T, \quad (28)$$

$$\kappa_k = \mathbf{P}_{x_k y_k} \mathbf{P}_{y_k y_k}^{-1}, \quad (29)$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + \kappa_k (y_k - \hat{y}_{k|k-1}), \quad (30)$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \kappa_k \mathbf{P}_{y_k y_k} \kappa_k^T. \quad (31)$$

where variables $\chi_{k|k-1}^a$, $\Upsilon_{k|k-1}$, $\hat{x}_{k|k-1}$, $\hat{y}_{k|k-1}$, Q_k , R_k , $\mathbf{P}_{k|k-1}$, $\mathbf{P}_{y_k y_k}$, $\mathbf{P}_{x_k y_k}$, κ_k , $\hat{x}_{k|k}$, $\mathbf{P}_{k|k}$ are prior sigma points states, prior output sigma points obtained from a nonlinear model given in

Equation (17), prior estimated state vector, prior estimated output vector, process noise covariance matrix, measured noise covariance matrix, prior state covariance matrix, prior output covariance matrix, cross covariance matrix, Kalman gain, posterior state vector—UKF output and a posteriori state covariance matrix, respectively. The UKF algorithm starts with the a priori selected weights W^m , W^c and scaling parameters λ , k_i . The UKF filter is used for the sensor fusion procedure in a manner to improve accuracy of the distance measurement with the Hall sensor. The direct measured value from the Hall sensor with Equation (14) represented variable y_k , where \hat{y}_k is the estimated variable from the derived nonlinear model of Equations (2), (3) and (13).

4.2. Deployment of the UKF Filter

The UKF filter, as a sensor fusion algorithm, is based on two main baselines; the real direct proximity measurement from the Hall sensor with nonlinear characteristic compensation given in Equation (14) and the proximity estimation from the selected non-linear dynamic model in Equations (2), (3) and (13). Both the proximity values are used in the UKF algorithm. The state space representation of the derived nonlinear model is:

$$\begin{aligned} \dot{x}(t) &= \begin{bmatrix} \dot{d}(t) \\ \dot{v}(t) \\ \dot{i}(t) \end{bmatrix} = \begin{bmatrix} v(t) \\ 9.89 - 5.996 \times 10^{-9} d^{-4}(t) + 4.509 \times 10^{-6} i(t) \cdot d^{-2}(t) \\ -165.6 i(t) + 61.61 u(t) \end{bmatrix} = \mathbf{F}(d(t), v(t), i(t)), \\ y(t) &= d(t) = \mathbf{H}(d(t), v(t), i(t)). \end{aligned} \quad (32)$$

where $x(t) = [d(t), v(t), i(t)]$ is a system state vector $x(t) \in \mathbb{R}^3$ and represents; d —PM vertical position, v —PM velocity and i —coil current. The size of the state vector is $L = 3$, which means that the UKF operated with $2L + 1$ sigma points. For the further use on a real-time system, the continuous-time Equation (32) were discretized using the Euler integration scheme. The discrete form of the continuous system in Equation (32) is:

$$\begin{aligned} d(k+1) &= d(k) + v(k) \cdot t_s, \\ v(k+1) &= v(k) + (g - 5.996 \times 10^{-9} d^{-4}(k) + 4.509 \times 10^{-6} i(k) \cdot d^{-2}(k)) \cdot t_s, \\ i(k+1) &= i(k) + (-165.6 i(k) + 61.61 u(k)) \cdot t_s, \\ y(k) &= d(k). \end{aligned} \quad (33)$$

where t_s is a sampling time with preselected value of, $t_s = 0.5$ ms. Model system output is the predicted proximity $d(k)$. For better transparency of the paper text, we assumed that the $d(k) = d_k$ and it applies for all other variables $v(k)$, $u(k)$, etc. The state variable \hat{x}_k of the UKF algorithm in regard to the Equations (22)–(31) is equal to $\hat{x}_k = [\hat{d}_k, \hat{v}_k, \hat{i}_k]^T$, where input vector u_k is a driven coil voltage. An important part of the algorithm is the state estimation update in Equation (30), which is based on error calculation between the direct proximity measurement d_{hall_k} obtained from Equation (14) and UKF's output state $\hat{d}_{k|k-1}$ given in Equations (23) and (25). The state update equation is, $\hat{x}_{k|k} = \hat{x}_{k|k-1} + \kappa_k (d_{hall_k} - \hat{d}_{k|k-1})$. The derived model in Equation (33) was used for sigma state estimation $\chi_{k|k-1}^a$ in Equation (22) and sigma output estimation $\Upsilon_{k|k-1}$ in Equation (25). The discrete process noise Q_k and measured noise R_k were set to:

$$\mathbf{Q}(k) = \begin{bmatrix} 5.6 \times 10^{-2} & 0 & 0 \\ 0 & 1.7 \times 10^{-2} & 0 \\ 0 & 0 & 7.71 \times 10^{-3} \end{bmatrix}, \mathbf{R}(k) = 0.12.$$

Other fixed parameters of the UKF have been selected arbitrary with values: $\alpha = 0.002$, $k_i = 0$, $\beta = 2$ [40]. The selected weighting matrices W^m and W^c are:

$$\begin{aligned} \mathbf{W}_m &= [-2.499 \ 0.416 \ 0.416 \ 0.416 \ 0.416 \ 0.416 \ 0.416] \times 10^5, \\ \mathbf{W}_c &= [-2.5 \ 0.416 \ 0.416 \ 0.416 \ 0.416 \ 0.416 \ 0.416] \times 10^5. \end{aligned}$$

Figure 12 represents deploying UKF as a sensor fusion algorithm for accurate proximity measurement from a Hall sensor.

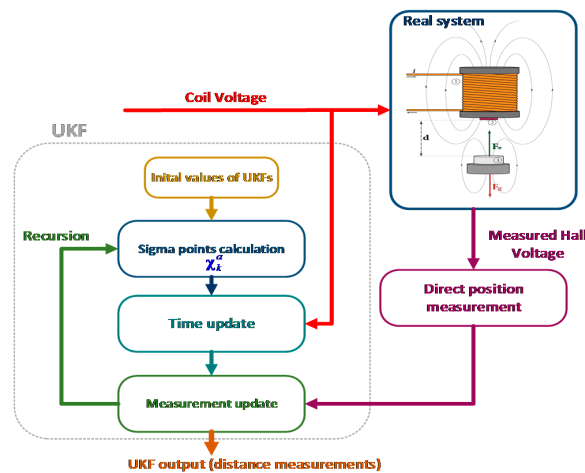


Figure 12. Sensor fusion algorithm with Unscented Kalman Filter for distance measurement of a levitating magnetic object.

5. Experimental Results

Testing the sensor fusion algorithm with a UKF filter was done predominantly in a closed loop system, with nonlinear and linear controllers. The nonlinear controller was obtained on the basis of backstepping controller design. The backstepping method is a recursive design technique, which stabilises the origin of the system in strict feedback form [44,45]. The nonlinear controller was synthesised from the derived model in Equation (32). A simplified second order nonlinear model in Equation (32) was used for the simplification of the Backstepping controller design. In the design procedure we assumed that the dynamic of the electrical coil was much higher than the mechanical part, where we got $\frac{di}{dt} = 0$. The static current value was $i(t) = \frac{61.61}{165.6} u(t)$. The static current value $i(t)$ was placed to the velocity equation $\frac{dv}{dt}$ of the model in Equation (32), wherein the second order model was obtained. For better reference tracking capability of the feedback system, the supplementary error state was introduced; $e(t) = ref(t) - d(t)$. The $ref(t)$ is a reference tracking value. The Backstepping controller procedure has had two 'back steps', from the proximity-error state $e(t)$ over $v(t)$ to the driven voltage as a system input and controller output. The nonlinear controller structure was:

$$u(k) = 596.11 \times 10^3 \cdot d(k) \cdot \left(\begin{array}{l} 9.83 + 0.3 \cdot ref_d(k) + ref_{dd}(k) - 3.45 \cdot v(k) + \\ 5.996 \times 10^{-9} d^{-4}(k) + ref(k) - d(k) \end{array} \right), \quad (34)$$

$$ref_d(k) = 0.927 \cdot ref_d(k-1) + 150 \cdot ref(k) - 150 \cdot ref(k-1),$$

$$ref_{dd}(k) = 0.9418 \cdot ref_{dd}(k-1) + 125 \cdot ref_d(k) - 125 \cdot ref_d(k-1),$$

where u_k , ref_k , ref_{dk} , ref_{ddk} are controller output-coil voltage, reference value, first derivative and second derivatives of the reference value, respectively. For the first and second derivatives a cut-off filter was used with frequency at the 150 Hz and 125 Hz. The velocity of the suspended object v_k was not measured directly with separated velocity sensor, but was estimated from a sensor fusion algorithm with UKF.

For the second test a classic linear PID structure with output clamping algorithm was used to validate the system efficiency with UKF. The PID controller was tuned with the linear approximation of the model given in Equation (32) and Integra of Square Error (ISE) performance index. The controller was designed with Control System Design and Optimization toolboxes in a MATLAB 2016a environment. The clamping algorithm was used to prevent proper operation of the system and prevent integrator windup. The PID controller structure with clamping output was:

$$u(k) = f(u(k)) \begin{pmatrix} 1.604 \cdot u(k-1) - 0.603 \cdot u(k-2) - 623.9 \cdot e(k) + \\ 1020 \cdot e(k-1) - 386.8 \cdot e(k-2) \end{pmatrix} + f_2(u(k)),$$

$$f_1(u(k)) = \begin{cases} 1, & 0.5 < u(k) < 3 \\ 0, & u(k) \leq 0.5 \\ 0, & u(k) \geq 3 \end{cases}, \quad f_2(u(k)) = \begin{cases} 0, & 0.5 < u(k) < 3 \\ 0.5, & u(k) \leq 0.5 \\ 3, & u(k) \geq 3 \end{cases}, \quad (35)$$

where u_k, e_k are controller output and current controller error, respectively.

The sensor UKF fusion algorithm and nonlinear controller are running on an ARM Cortex-M4 STM32F407VGT6 microcontroller (STMicroelectronics, Geneva, Switzerland) with floating point unit-FPU and 12 bit AD conversion for accurate Hall voltage measurement. Figure 13 represents a real-time system configuration with a Hall proximity sensor and ARM microcontroller. The schematic in Figure 14 represents the structure of the sensor fusion algorithm with the feedback controller in Figure 14a and the algorithm procedure flow chart in Figure 14b.

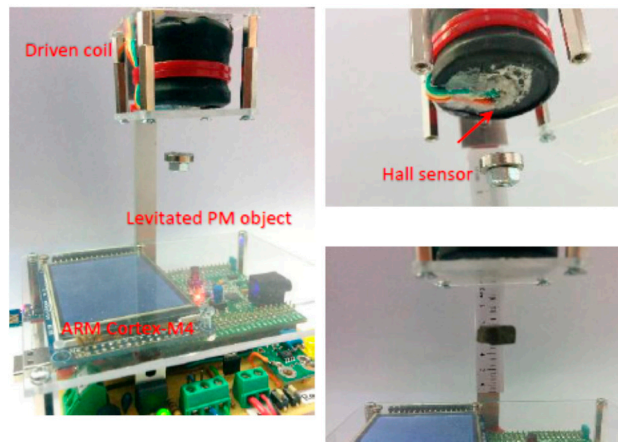


Figure 13. The real-time system with UKF-Hall positioning sensor and feedback control.

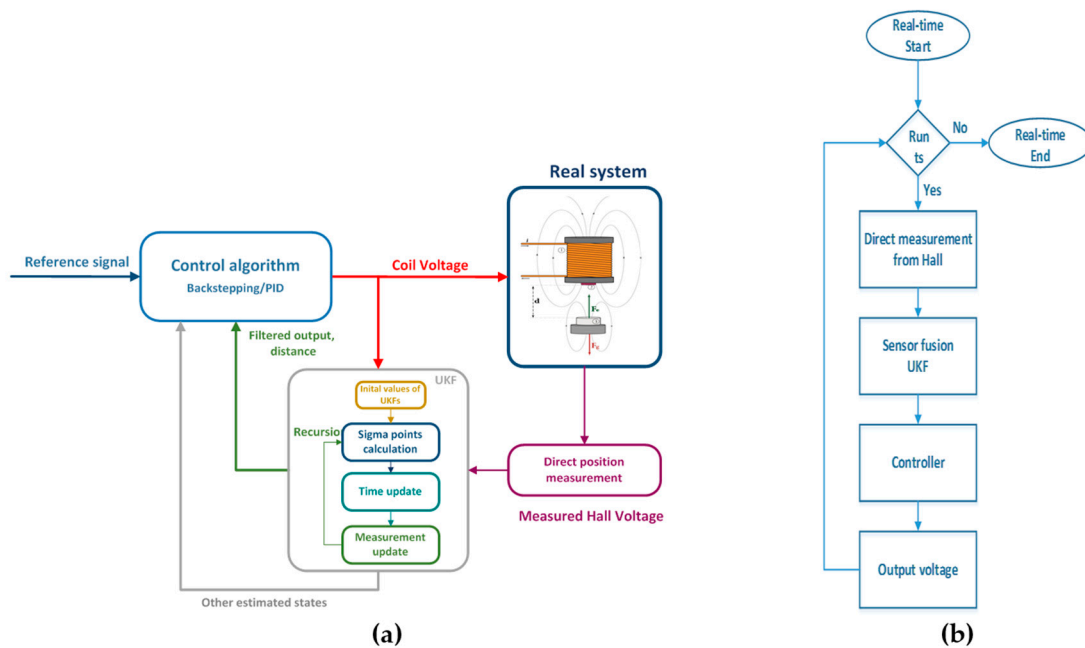


Figure 14. (a) Schematic of an algorithm structure and (b) Flow chart of real time algorithm execution.

Figures 15–18 represent the experimental results of the sensor fusion algorithm with UKF and direct measurements distance with static characteristics (Equation (14)). The reference value of the feedback system represents the true proximity of the PM obtained from the external independent ruler. The accuracy of the external ruler was around 0.2 mm. The experiments were tested with feedback controllers (backstepping and PID), where feedback signals (position, velocity) are taken from the UKF sensor fusion algorithm. In the presented results, the reference value was changed in the span of 22 mm to the 30 mm from the driven coil.

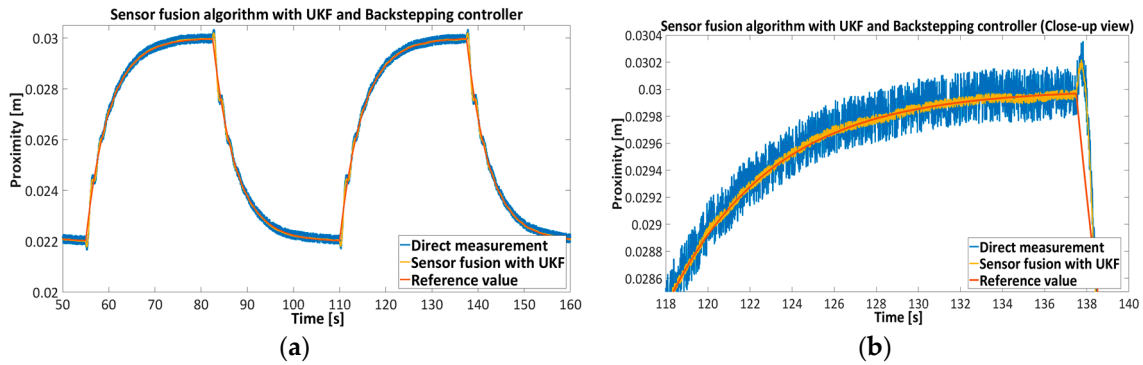


Figure 15. Feedback control of EMaWS with nonlinear Backstepping controller, comparison between direct proximity measurement and sensor fusion-UKF proximity algorithm.

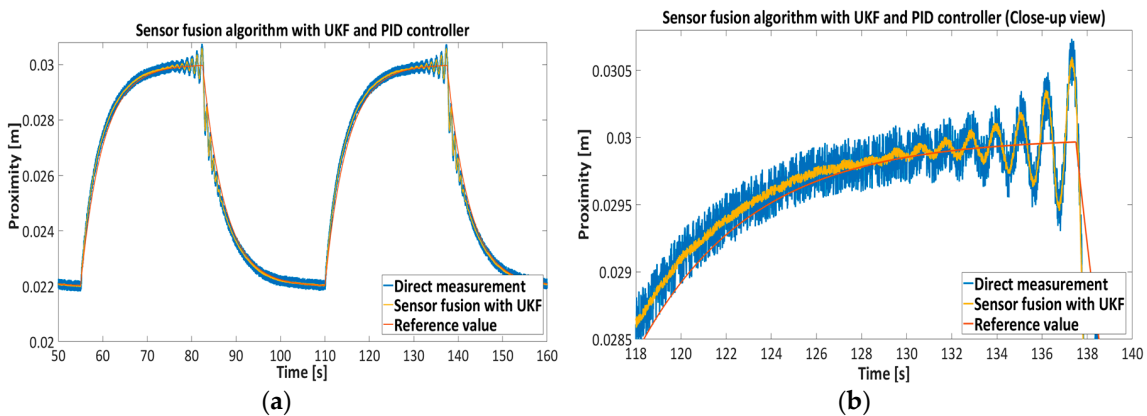


Figure 16. Feedback control of EMaWS with PID controller, comparison between direct proximity measurement and sensor fusion-UKF proximity algorithm.

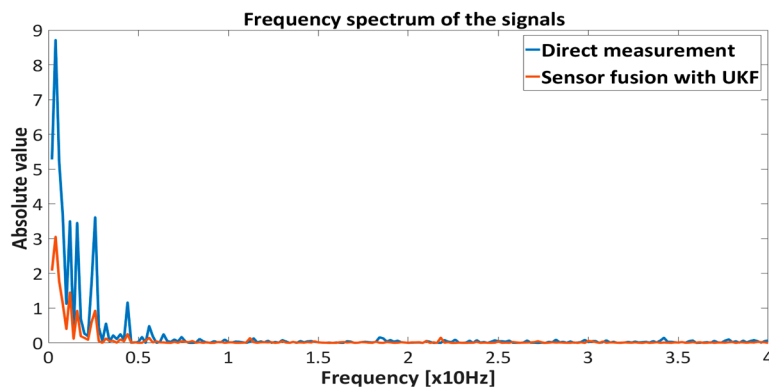


Figure 17. Frequency spectrum of the proximity measurements (direct measurement and sensor fusion) with Backstepping controller.

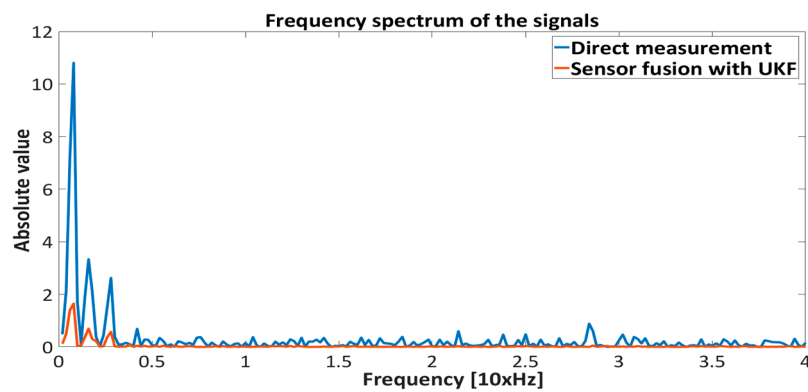


Figure 18. Frequency spectrum of the proximity measurements (direct measurement and sensor fusion) with PID controller.

Table 6 shows the Root Mean Square-RMS value of the measurements with Backstepping and PID controllers, where RMS value was assessed with the given expression, $RMS = \sqrt{\frac{1}{N} \sum_{k=1}^N (d_k - \hat{d}_k)^2}$ where d_k is a true value and \hat{d}_k is the measurement from direct approximation or sensor fusion algorithm with UKF.

Table 6. RMS values of the signals.

RMS	Backstepping Controller	PID Controller
Direct measurement	12.32×10^{-5}	15.71×10^{-5}
Sensor fusion with UKF	1.52×10^{-5}	1.68×10^{-5}

Figure 19 presents a comparison between the UKF and EKF sensor fusion algorithms with the backstepping controller given in Equation (34). The EKF algorithm uses a linear approximation (Jacobian matrix) of the model in Equation (32). For better comparison of both algorithms the span of reference value was selected between 22 mm and 25 mm.

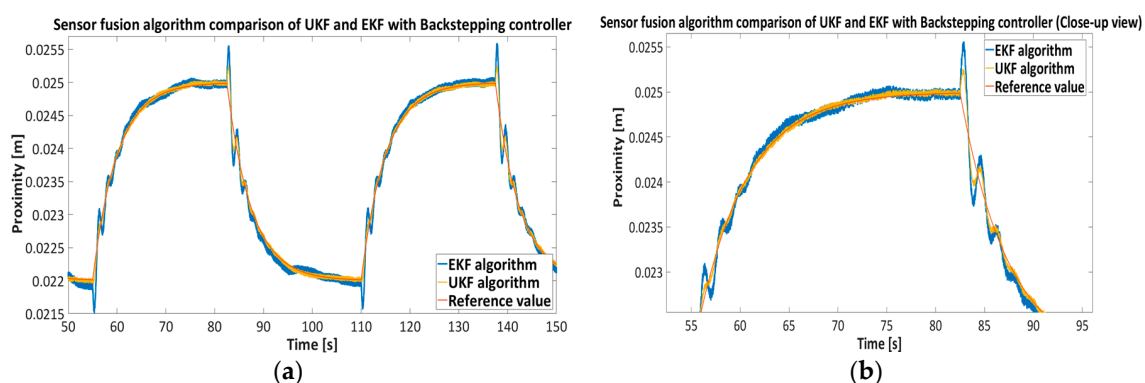


Figure 19. Comparison of the UKF and EKF sensor fusion algorithms with nonlinear backstepping feedback controller.

The results presented in Figures 15–18 show the effectiveness and reliability of the sensor fusion algorithm with UKF. The UKF, in regard to the direct measurement application improves proximity measurements significantly, suppresses Hall drift and lowers noise influence on the proximity information signal. The effectiveness of the noise suppression is confirmed with the Frequency spectrum plot in Figures 17 and 18 and with Table 6, where the RMS values of the signals are calculated.

From Figure 19 it can be seen clearly that the UKF sensor fusion algorithm outperforms the EKF. The advantage of UKF can be noticed also in its reference tracking and noise suppression capability. The resolution of the measurement in regard to the feedback stability region between 18 mm and 32 mm is estimated at 0.05 mm with regard to Equations (14)–(16), Hall sensitivity value and 12 bit AD resolution of the ARM microcontroller. The accuracy of the proximity measurement in regard to the external ruler is estimated at 0.2 mm. At the end of the experiment it needs to be mentioned that the close-loop system in both configurations with backstepping and PID controllers are unstable if direct measurement from the Hall sensors is used in controller operation.

6. Conclusions

The main contribution of the presented paper is accurate distance measurement with low cost sensing devices in the presence of a magnetic field. In the present case a low cost linear Hall sensor was used. The rough data from the sensor are relatively noisy and contain exogenous disturbance effects, which need to be removed or suppressed efficiently. The paper findings show great distance measurement results with regard to the system open loop instability and feedback controller sensitivity to the sensor noise and signal drift. The sensor fusion algorithm with UKF improved the accuracy of distance measurement and system states estimation drastically, which are also used in feedback control. The efficiency of the sensor fusion algorithm originated from the accuracy of the system model derived with the Finite Element Method. Both approaches, modelling of the system dynamics with the Finite Element Method and UKF filter settings, are crucial in the proximity measuring with a Hall sensor. Such an approach can be used in many different electro-mechanical applications where a relatively uncertain sensor is used and system behavior is known. The approach offers a great potential to acquiring the quantities, which are not directly measured with separated sensors but are estimated with the model and sensor fusion algorithm (measuring the coil current and the velocity of the levitating object).

Acknowledgments: This work is supported by the Margento R&D Company and the Faculty of Electrical Engineering and Computer Science, University of Maribor.

Author Contributions: Amor Chowdhury developed the methodology of the modelling with a FEM approach. Both authors derived sensor fusion algorithm and drafted the manuscript. Andrej Sarjaš implemented and prepared the Hardware experiments.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Popovic, R.S. *Hall Effect Devices*, 2nd ed.; CRC Press-Taylor & Francis Group: Boca Raton, FL, USA, 2003.
2. Ramsden, E. *Hall-Effect Sensors: Theory and Applications*, 2nd ed.; Elsevier: Amsterdam, The Netherlands; Newnes: Amsterdam, The Netherlands, 2006.
3. Yan, L.; Zhu, B.; Jiao, Z.X.; Chen, C.Y.; Chen, I.M. An orientation measurement method based on Hall-effect sensors for permanent magnet spherical actuators with 3d magnet array. *Sci. Rep.* **2014**. [[CrossRef](#)] [[PubMed](#)]
4. Yan, L.; Zhu, B.; Jiao, Z.X.; Chen, C.Y.; Chen, I.M. Hall-sensor-based orientation measurement method in three-dimensional space for electromagnetic actuators. In Proceedings of the 2014 IEEE International Conference on Automation Science and Engineering (CASE), New Taipei, Taiwan, 18–22 August 2014; pp. 182–187.
5. Krause, P.C.; Wasynczuk, O.; Sudhoff, S.D. *Analysis of Electric Machinery and Drive Systems*; IEEE Press: Piscataway, NY, USA, 2002.
6. Fontana, M.; Salsedo, F.; Bergamasco, M. Novel magnetic sensing approach with improved linearity. *Sensors* **2013**, *13*, 7618–7632. [[CrossRef](#)] [[PubMed](#)]
7. Li, J.H.; Chiou, J.S. Digital control analysis and design of a field-sensed magnetic suspension system. *Sensors* **2015**, *15*, 6174–6195. [[CrossRef](#)] [[PubMed](#)]
8. Li, J.H.; Chiou, J.S. GSA-tuning IPD control of a field-sensed magnetic suspension system. *Sensors* **2015**, *15*, 31781–31793. [[CrossRef](#)] [[PubMed](#)]

9. Lin, C.M.; Lin, M.H.; Chen, C.W. SoPC-based adaptive PID control system design for magnetic levitation system. *IEEE Syst. J.* **2011**, *5*, 278–287. [[CrossRef](#)]
10. Mehrtash, M.; Khamesee, M.B. Design and implementation of LQG/LTR controller for a magnetic telemanipulation system—Performance evaluation and energy saving. *J. Microsyst. Technol.* **2011**, *14*, 1135–1143. [[CrossRef](#)]
11. Shameli, E.; Khamesee, M.B.; Huissoon, J.P. Nonlinear controller design for a magnetic levitation device. *Microsyst. Technol.* **2007**, *13*, 831–835. [[CrossRef](#)]
12. Elbuken, C.; Khamesee, M.B.; Yavuz, M. Design and implementation of a micromanipulation system using a magnetically levitated MEMS robot. *IEEE/ASME Trans. Mechatron.* **2009**, *14*, 434–445. [[CrossRef](#)]
13. An, S.; Ma, Y.; Cao, Z. Applying simple adaptive control to magnetic levitation system. In Proceedings of the Second International Conference on Intelligent Computation Technology and Automation, Changsha, China, 10–11 October 2009; pp. 746–749.
14. Lin, F.J.; Chen, S.Y.; Shyu, K.K. Robust dynamic sliding-mode control using adaptive RENN for magnetic levitation system. *IEEE Trans. Neural Netw.* **2009**, *20*, 938–951. [[PubMed](#)]
15. Yang, Z.J.; Kunitoshi, K.; Kanae, S.; Wada, K. Adaptive robust output-feedback control of a magnetic levitation system by K-filter approach. *IEEE Trans. Ind. Electron.* **2008**, *55*, 390–399. [[CrossRef](#)]
16. Gentili, L.; Marconi, L. Robust nonlinear disturbance suppression of a magnetic levitation system. *Automatica* **2003**, *39*, 735–742. [[CrossRef](#)]
17. Kashif, I.; Yasir, S.; Abdullah, S.S.; Amjad, M.; Munaf, R.; Suhail, K. Modeling and control of magnetic levitation system via fuzzy logic controller. In Proceedings of the 4th International Conference on Modeling, Simulation and Applied Optimization (ICMSAO), Kuala Lumpur, Malaysia, 19–21 April 2011. [[CrossRef](#)]
18. Castanedo, F. A review of data fusion techniques. *Sci. World J.* **2013**, *2013*, 704504. [[CrossRef](#)] [[PubMed](#)]
19. Kalman, R.E. A new approach to linear filtering and prediction problems. *J. Basic Eng.* **1960**, *82*, 35–45. [[CrossRef](#)]
20. Lei, H. Auto regressive moving average (ARMA) modeling method for Gyro random noise using a robust Kalman filter. *Sensors* **2015**, *15*, 25277–25286.
21. Evensen, G. *Data Assimilation: The Ensemble Kalman Filter*; Springer: Berlin, Germany, 2007.
22. Naumović, M.B. Modeling of a didactic magnetic levitation system for control education. In Proceedings of the International Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Services—TELSIKS2003, Niš, Serbia and Montenegro, 1–3 October 2003; pp. 783–786.
23. Hajjaji, E.A.; Ouladsine, M. Modeling and nonlinear control of magnetic levitation system. *IEEE Trans. Ind. Electron.* **2001**, *48*, 831–838. [[CrossRef](#)]
24. Qin, Y.M.; Peng, H.; Ruan, W.J.; Wu, J.; Gao, J.C. A modeling and control approach to magnetic levitation system based on state-dependent ARX model. *J. Process. Control* **2014**, *24*, 93–112. [[CrossRef](#)]
25. Shameli, E.; Khamesee, M.B.; Huissoon, J.P. Frequency response identification and dynamic modeling of a magnetic levitation device. In Proceedings of the ASME 2007 International Mechanical Engineering Congress and Exposition American Society of Mechanical Engineers, Seattle, Washington, DC, USA, 11–15 November 2007; pp. 1635–1639.
26. Shameli, E.; Khamesee, M.B.; Huissoon, J.P. Real-time control of a magnetic levitation device based on instantaneous modeling of magnetic field. *Mechatronics* **2008**, *18*, 536–544. [[CrossRef](#)]
27. Coulomb, J.; Meunier, G. Finite element implementation of virtual work principle for magnetic or electric force and torque computation. *IEEE Trans. Magn.* **1984**, *20*, 1894–1896. [[CrossRef](#)]
28. Meessen, K.J.; Paulides, J.J.H.; Lomonova, E.A. Force calculations in 3-D cylindrical structures using fourier analysis and the Maxwell stress tensor. *IEEE Trans. Magn.* **2013**, *49*, 536–545. [[CrossRef](#)]
29. Griffiths, D.J. *Introduction to Electrodynamics*, 3rd ed.; Prentice Hall: New Jersey, NJ, USA, 1998.
30. Naumović, M.B. Nonlinear state observation in a didactic magnetic levitation system. In Proceedings of the International Scientific Conference on Information, Communication and Energy Systems and Technologies (ICEST 2004), Bitola, Macedonia, 16–19 June 2004; pp. 473–476.
31. Furlani, E.P. *Permanent Magnet and Electromechanical Devices: Materials, Analysis and Applications*; Academic Press: New York, NY, USA, 2001.
32. Robertson, W.; Cazzolato, B.; Zander, A. Axial force between a thick coil and a cylindrical permanent magnet: Optimizing the geometry of an electromagnetic actuator. *IEEE Trans. Magn.* **2012**, *48*, 536–545. [[CrossRef](#)]
33. Reddy, J.N. *An Introduction to the Finite Element Method*, 3rd ed.; McGraw-Hill: Singapore, 2006.

34. Jin, J. *The Finite Element Method in Electromagnetics*; John Wiley and Sons: New York, NY, USA, 1993.
35. Meeker, D. FEMM42.exe Software. Available online: <http://www.femm.info> (accessed on 6 August 1998).
36. Benamimour, T.; Bentounsi, A.; Djeghloud, H. CAD of electrical machines using coupled FEMM-MATLAB softwares. In Proceedings of the 3rd International Conference on Electric Power and Energy Conversion Systems (EPECS), Istanbul, Turkey, 2–4 October 2013; pp. 1–6.
37. Gould, N.; Toint, P.L. Preprocessing for quadratic programming. *Math. Program.* **2004**, *100*, 95–132. [[CrossRef](#)]
38. Sabatini, A.M. Variable-state-dimension Kalman-based filter for orientation determination using inertial and magnetic sensors. *Sensors* **2012**, *12*, 8491–8506. [[CrossRef](#)] [[PubMed](#)]
39. Ligorio, S.; Sabatini, A.M. Extended Kalman filter-based methods for pose estimation using visual, inertial and magnetic sensors: Comparative analysis and performance evaluation. *Sensors* **2013**, *13*, 1919–1941. [[CrossRef](#)] [[PubMed](#)]
40. Wan, A.E.; Van der Merwe, R. The unscented Kalman filter for nonlinear estimation. In Proceedings of the Adaptive Systems for Signal Processing, Communications, and Control Symposium, Lake Louise, Canada, 1–4 October 2000; pp. 153–158.
41. Julier, S.J. The scaled unscented transformation. In Proceedings of the American Control Conference, Anchorage, KY, USA, 8–10 May 2002; pp. 4555–4559.
42. Julier, S.J.; Uhlmann, J.K. Unscented filtering and nonlinear estimation. *Proc. IEEE* **2004**, *92*, 401–422. [[CrossRef](#)]
43. Särkkä, S. On Unscented Kalman filtering for state estimation of continuous-time nonlinear systems. *IEEE Trans. Autom. Control* **2007**, *52*, 1631–1641. [[CrossRef](#)]
44. Kokotovic, P.V. The joy of feedback nonlinear and adaptive. *Control Syst. Mag.* **1992**, *12*, 7–17. [[CrossRef](#)]
45. Galeazzi, R. Observer backstepping control for variable speed wind turbine. In Proceedings of the American Control Conference, Washington, WA, USA, 17–19 June 2013; pp. 1036–1043.



© 2016 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).