

Article

A Robust Static Decoupling Algorithm for 3-Axis Force Sensors Based on Coupling Error Model and ϵ -SVR

Junqing Ma ¹, Aiguo Song ^{1,*} and Jing Xiao ²

¹ Jiangsu Key Lab of Remote Measurement and Control, School of Instrument Science and Engineering, Southeast University, Nanjing 210096, China; E-Mail: mjqsou@gmail.com

² IMI Lab, Department of Computer Science, The University of North Carolina at Charlotte, Charlotte, NC 28223, USA; E-Mail: xiao@uncc.edu

* Author to whom correspondence should be addressed; E-Mail: a.g.song@seu.edu.cn; Tel.: +86-25-8379-3293; Fax: +86-25-8379-2813.

Received: 14 August 2012; in revised form: 22 October 2012 / Accepted: 22 October 2012 /

Published: 29 October 2012

Abstract: Coupling errors are major threats to the accuracy of 3-axis force sensors. Design of decoupling algorithms is a challenging topic due to the uncertainty of coupling errors. The conventional nonlinear decoupling algorithms by a standard Neural Network (NN) are sometimes unstable due to overfitting. In order to avoid overfitting and minimize the negative effect of random noises and gross errors in calibration data, we propose a novel nonlinear static decoupling algorithm based on the establishment of a coupling error model. Instead of regarding the whole system as a black box in conventional algorithm, the coupling error model is designed by the principle of coupling errors, in which the nonlinear relationships between forces and coupling errors in each dimension are calculated separately. Six separate Support Vector Regressions (SVRs) are employed for their ability to perform adaptive, nonlinear data fitting. The decoupling performance of the proposed algorithm is compared with the conventional method by utilizing obtained data from the static calibration experiment of a 3-axis force sensor. Experimental results show that the proposed decoupling algorithm gives more robust performance with high efficiency and decoupling accuracy, and can thus be potentially applied to the decoupling application of 3-axis force sensors.

Keywords: force sensors; coupling errors; decoupling; support vector regression (SVR)

1. Introduction

Force sensing is crucial for on-line perception and feedback in interactions between intelligent robotic manipulators and environments. Multi-axis force sensors are used to perceive generalized force information and convert input force signals to voltage signals [1]. They are usually mounted on the tips of robot arms for automatic contact recognition, motion planning, and force control tasks [2–4]. Force sensors are widely used in assembly robots, polishing robots, rehabilitation robots, *etc.* [5–7].

The accuracy of multi-axis force sensors has a great impact on force-perception based tasks with high precision requirements. This motivates the need to improve measurement precision. For a multi-axis force sensor, a key issue is that input force in one dimension may affect not only output of this dimension but also those of the other dimensions. Errors caused in this way, called the *coupling errors*, are major threats to the accuracy of multi-axis force sensors. Coupling errors occur for various reasons, such as mechanical structures, limitation of machining accuracy, transverse effect of strain gauges, *etc.* Song *et al.* in [8] developed a self-decoupled 4-axis force/torque sensor to reduce coupling errors by improving hardware design. However, in most cases, it is costly and sometimes infeasible to avoid coupling errors by improving the hardware design and machining accuracy. Decoupling algorithms are always used to reduce coupling errors.

The common static decoupling algorithm calculates the pseudo-inverse matrix of calibration data based on the Least Square Method (LSM) [9–11]. This algorithm is based on the assumption that relationships between input forces and output voltages in all dimensions are linear. Afterwards, the transfer matrix between input forces and output voltages are calculated. The obtained transfer matrix is called *calibration matrix*. Voyles *et al.* in [12] proposed a fast linear decoupling technique called *shape from motion* in which the motion of the force vector and the calibration matrix are simultaneously extracted by singular value decomposition from raw sensor signals. Cao *et al.* in [13] explored a linear static decoupling method using an NN to increase the accuracy of decoupling. However, large amounts of experiment data indicate the nonlinearity in relationships between forces and coupling errors. Thus, the precision of linear decoupling algorithms is limited and unsatisfactory. Other approaches [14,15] employed a feed-forward NN with back propagation (BP) training algorithms to realize the nonlinear Multiple Input Multiple Output (MIMO) mapping of a multi-axis force sensor. In [15], the authors also used a standard radial basis function (RBF) NN for decoupling. Engineering applications show that decoupling algorithms with a standard NN model can sometimes reduce coupling error significantly, but sometimes generate worse results than without decoupling due to overfitting.

Support Vector Machine (SVM) is a powerful candidate for decoupling algorithms due to its ability to perform adaptive and nonlinear data fitting. SVM starts from solving problems of classification. With the introduction of Vapnik's ϵ -insensitive loss function, it also extends to be a regression prediction tool that uses machine learning theory to maximize predictive accuracy while not subject to local minimal and overfitting [16]. Support Vector Machine for regression, called *Support Vector Regression*, gradually becomes a powerful tool for nonlinear correcting and compensation in the field of sensors. Guo *et al.* in [17] established a model based on SVR to correct the nonlinear error of photoelectric displacement sensor. Wang in [18] used SVR to make nonlinear estimation and temperature compensation of capacitor pressure sensors.

The design of decoupling algorithms of 3-axis force sensors presents several challenges. First, a 3-axis force sensor is usually used in on-line force perception tasks. This requires the sensor to show quick response to variations of input forces. Thus, the decoupling algorithms should have high time efficiency. Second, inevitable noises in calibration data may result in overfitting such that decoupling precision will be reduced. Also, due to improper operations of laboratory technicians or environment disturbances, occasionally there may be gross errors in calibration data. Gross errors are outliers that strongly deviate from the majority of experiment data. Gross errors may result in unexpected decoupling results such that decoupling precision will be reduced. A reliable decoupling algorithm should be designed to minimize the negative effects of noises and gross errors. Third, because the hardware causes of coupling errors such as the mechanical structures and the limitations of machining accuracy are complex and uncertain, it is difficult to model the coupling errors by polynomials. Thus, a decoupling algorithm should have a high generalization ability.

Motivated by the above challenges, this paper proposes a precise and fast decoupling algorithm with high reliability. Instead of referring to the whole sensor system as a black box using one standard NN [14,15], the proposed decoupling algorithm is designed using the principle of coupling errors, in which the relationships between each input and output are mapped separately in the proposed coupling error model to make the algorithm more reliable. The proposed coupling error model consists of six SVRs and three linear fitting functions, which is more conformable to calibration data structure. Our method is compared with the standard NN method, as they are applied to the same data from a calibration experiment, and our method gives better reliability and higher efficiency.

The remainder of the paper is organized as follows. Section 2 introduces a novel model of coupling error and its notations. ϵ -SVR is described as the nonlinear approximation tool of the model. A decoupling process based on the model is proposed. Section 3 briefly describes the principle of a 3-axis sensor designed in our lab, the calibration experiment process and the structure of the calibration data. Section 4 discusses the implementation details of the decoupling method using a set of experimental data obtained from a calibration experiment. In order to demonstrate that our decoupling algorithm is robust to gross errors, gross errors were artificially introduced into the calibration data and the decoupling accuracy with the gross errors were calculated. Finally, in Section 5 some conclusions are provided.

2. Nonlinear Static Decoupling

2.1. Coupling Error Model and Notations

We first establish an appropriate coupling error model to capture the relationships between input forces and corresponding coupling errors. In the model, the input forces and output voltages of a 3-axis force sensor in X, Y, Z directions are defined as f_x, f_y, f_z and u_x, u_y, u_z , respectively.

For each dimension, output voltages are partitioned into two categories. One category includes the voltages corresponding to input forces in the same dimension, called *prime voltages*. The other category includes the voltages corresponding to the input forces in the other two dimensions, called *coupling errors*. We use u_{xx}, u_{yy}, u_{zz} to denote prime voltages and e_x, e_y, e_z to denote coupling errors in X, Y, Z directions, respectively. Prime voltages account for the majority of output voltages. Next, coupling

errors are separated into two *coupling error elements* caused by input forces of different dimensions. Let (e_{xy}, e_{xz}) represent the coupling error element in X direction, where e_{xy} refers to the coupling error element caused by f_y , and e_{xz} refers to the coupling error element caused by f_z . Similarly, we split the coupling error in Y direction into e_{yx} and e_{yz} , and split the coupling error in Z direction into e_{zx} and e_{zy} . We can get:

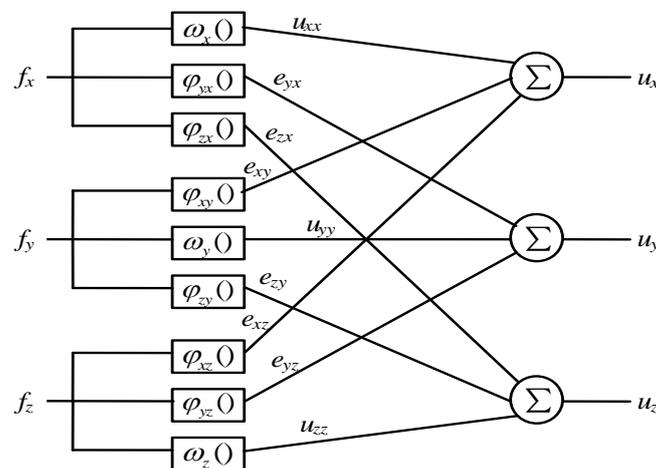
$$\begin{cases} u_x = u_{xx} + e_{xy} + e_{xz} \\ u_y = u_{yy} + e_{yx} + e_{yz} \\ u_z = u_{zz} + e_{zx} + e_{zy} \end{cases} \quad (1)$$

Based on the observation of calibration data of multi-axis force sensors in our lab, we make the following assumptions about coupling errors.

- (1) The relationship between the prime force and the prime voltage in every dimension is linear;
- (2) Relationships between disturbing force and their corresponding coupling error elements in every dimension are nonlinear;
- (3) The above relationships are independent and time-invariant.

From the above assumptions and the principle of superposition for stress, we propose a coupling error model as shown in Figure 1.

Figure 1. A coupling error model of 3-axis force sensors.



In Figure 1, there are three layers in the coupling error model: the input layer, the output layer, and the middle layer. Nine nodes in the middle layer are parallel and separated from each other. Functions $\omega_x()$, $\omega_y()$ and $\omega_z()$, are *non-coupling functions*, linearly relating prime force to prime voltage in each dimension as shown in Equation (2).

$$\begin{cases} u_{xx} = \omega_x(f_x) = k_{xx} \cdot f_x \\ u_{yy} = \omega_y(f_y) = k_{yy} \cdot f_y \\ u_{zz} = \omega_z(f_z) = k_{zz} \cdot f_z \end{cases} \quad (2)$$

In calibration experiments, only one-dimensional force is applied to a 3-axis force sensor each time, while the output voltages of all directions are recorded simultaneously. Detailed calibration experiment process will be described in Section 3. Consequently, as for calibration data, no coupling error exists in the output voltage corresponding to the direction of the input force and no prime force exists in the output voltages of other directions. In other words, the output voltage equals the prime voltage when the direction of the output voltage is the same as the direction of the input force, and the output voltage equals the corresponding coupling error elements when the direction of the output voltage is different from the direction of the input force. For instance, during calibration experiment of X direction, a set of standard $f_{x(c)}$ are applied to the force sensor while $f_{y(c)}$ and $f_{z(c)}$ remain zero (the subscript c represents “calibration data”). Thus we can get $u_{x(c)} = u_{xx(c)}$, $u_{y(c)} = e_{yx(c)}$, $u_{z(c)} = e_{zx(c)}$. Similarly, for calibration data in Y direction, $u_{x(c)} = e_{xy(c)}$, $u_{y(c)} = u_{yy(c)}$, $u_{z(c)} = e_{zy(c)}$; for calibration data in Z direction, $u_{x(c)} = e_{xz(c)}$, $u_{y(c)} = e_{yz(c)}$, $u_{z(c)} = u_{zz(c)}$. Thus, coefficients k_{xx} , k_{yy} , and k_{zz} in Equation (2) can be calculated by linear fitting of prime forces and prime voltages of calibration data using LSM.

Functions respecting nonlinear relationships between disturbing forces and corresponding coupling error elements in every dimension are $\varphi_{yx}()$, $\varphi_{zx}()$, $\varphi_{xy}()$, $\varphi_{zy}()$, $\varphi_{xz}()$, $\varphi_{yz}()$, called *coupling functions* as shown in Equation (3).

$$\begin{cases} e_{yx} = \varphi_{yx}(f_x); e_{zx} = \varphi_{zx}(f_x); \\ e_{xy} = \varphi_{xy}(f_y); e_{zy} = \varphi_{zy}(f_y); \\ e_{xz} = \varphi_{xz}(f_z); e_{yz} = \varphi_{yz}(f_z); \end{cases} \quad (3)$$

Then, *corrected coupling functions* $\chi()$ can be obtained from Equations (2) and (3):

$$\begin{cases} e_{yx} = \varphi_{yx}(f_x) = \varphi_{yx}(\omega_x^{-1}(u_{xx})) = \chi_{yx}(u_{xx}) \\ e_{zx} = \varphi_{zx}(f_x) = \varphi_{zx}(\omega_x^{-1}(u_{xx})) = \chi_{zx}(u_{xx}) \\ e_{xy} = \varphi_{xy}(f_y) = \varphi_{xy}(\omega_y^{-1}(u_{yy})) = \chi_{xy}(u_{yy}) \\ e_{zy} = \varphi_{zy}(f_y) = \varphi_{zy}(\omega_y^{-1}(u_{yy})) = \chi_{zy}(u_{yy}) \\ e_{xz} = \varphi_{xz}(f_z) = \varphi_{xz}(\omega_z^{-1}(u_{zz})) = \chi_{xz}(u_{zz}) \\ e_{yz} = \varphi_{yz}(f_z) = \varphi_{yz}(\omega_z^{-1}(u_{zz})) = \chi_{yz}(u_{zz}) \end{cases} \quad (4)$$

During a decoupling process of an actual force perception task, one can hardly obtain the exact value of prime voltages u_{xx} , u_{yy} , u_{zz} for Equation (4), because the data obtained in a force perception task (called *task data*) are different from calibration data. The input forces in all directions are always non-zero, so the output voltages in all directions contain coupling errors and the output voltages no longer equal the prime voltages. However, a large number of experiment data show that, in most cases, coupling errors take up less than 5% of the full scale (F.S.) output voltages and the absolute values of the slopes of the corrected coupling functions are no more than 0.05. Thus, the output voltages of task data $u_{x(t)}$, $u_{y(t)}$, $u_{z(t)}$ approximately equal the prime voltages $u_{xx(t)}$, $u_{yy(t)}$, $u_{zz(t)}$ (the subscript t represents “task data”), and can be used as substitutions of the prime voltages $u_{xx(t)}$, $u_{yy(t)}$, $u_{zz(t)}$ as independent variables of Equation (4). This approximate substitution may induce second-order coupling errors. Take

e_{yx} for example: the measured coupling error element $e_{yx(m)} = \chi_{yx}(u_{x(t)})$ is calculated instead of the actual coupling error element $e_{yx(a)} = \chi_{yx}(u_{xx(t)})$ (the subscript m represents “measured”; the subscript a represents “actual”). The second-order coupling error $E_{e_{yx}}$ can be expressed in Equation (5),

$$E_{e_{yx}} = \frac{|e_{yx(m)} - e_{yx(a)}|}{u_{x(F.S.)}} = \frac{|\chi_{yx}(u_{x(t)}) - \chi_{yx}(u_{xx(t)})|}{u_{x(F.S.)}} \quad (5)$$

where $u_{x(F.S.)}$ represents the full scale value of output voltages in X direction.

According to the Lagrange mean value theorem, $\exists \xi \in (u_{xx(t)}, u_{x(t)})$ such that

$$E_{e_{yx}} = \frac{|\chi'_{yx}(\xi)| |u_{x(t)} - u_{xx(t)}|}{u_{x(F.S.)}} \leq \frac{0.05 \cdot (5\% u_{x(F.S.)})}{u_{x(F.S.)}} = 0.25\% \quad (6)$$

As a result, the rate of second-order coupling errors is less than 0.25% and can be neglected in most industrial applications.

We choose ϵ -SVR for nonlinear approximations for the corrected coupling functions, which is much less likely to subject to overfitting problems. Also, the fitting processes of those functions are independent from one dimension to another, hence the fitting result of one dimension does not affect that of another dimensions.

2.2. Approximation of Corrected Coupling Functions Using ϵ -SVR

Six ϵ -SVRs are utilized to approximate six corrected coupling functions for its generalization ability. ϵ -SVR learns the relationship between the input (*i.e.*, prime voltages) and the output (*i.e.*, corresponding coupling error elements) by adjusting the structure and parameters of a flexible model directly from training data to minimize the prediction error.

The basic idea for the case of nonlinear regression by ϵ -SVR is to project the input space x_i to a higher dimensional feature space by a map Φ . Then, the ϵ -SVR defines a linear prediction model over the mapped samples in the feature space. A nonlinear function is learned by this model while the capacity of the model is controlled by a parameter that does not depend on the dimensionality of the space [16]. As calculation with the map Φ can easily become computationally infeasible (because it is too complex), a kernel function k is introduced as the dot product of Φ , as expressed in Equation (7).

$$\langle \Phi(x_i), \Phi(x_j) \rangle = k(x_i, x_j) \quad (7)$$

Typical kernels include the linear $k(x_i, x_j) = x_i^T x_j$, the polynomial $k(x_i, x_j) = (x_i^T x_j + 1)^d$, and the Gaussian Function (RBF) $k(x_i, x_j) = \exp(-\|x_i - x_j\|^2 / 2\delta^2)$. The RBF kernel is most frequently used and it is also the one used in our implementation.

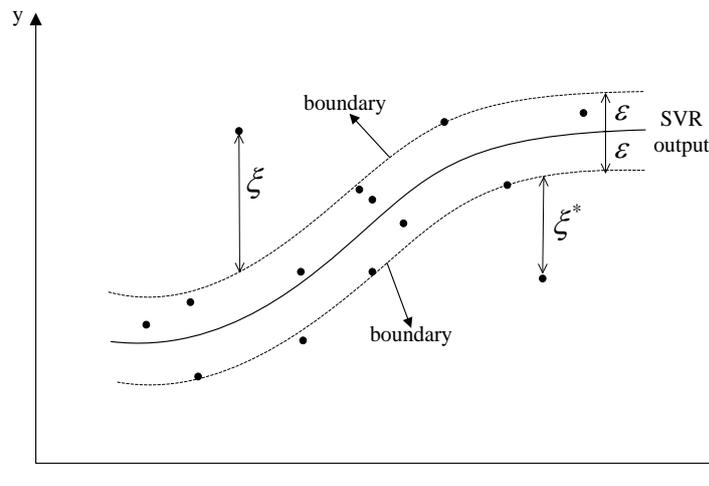
Briefly, the ϵ -SVR is to solve a convex optimization problem:

$$\text{minimize } \frac{1}{2} \|w\|^2 + C \sum_{i=1}^Q (\xi_i + \xi_i^*) \quad (8a)$$

$$\text{subject to } \begin{cases} y_i - w\Phi(x_i) - b \leq \epsilon + \xi_i \\ w\Phi(x_i) + b - y_i \leq \epsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases} \quad (8b)$$

where $i = 1, 2, \dots, Q$ are training data points, the parameter w indicates the flatness of regression function f due to the fact that kernels can be associated with flatness properties via regularization operators [19]. ξ_i, ξ_i^* are slack variables introduced by Vapnik's ϵ -insensitive loss function [20], in which errors up to ϵ are not penalized, and all further deviations will incur a linear penalization [21]. $C > 0$ is the regularization parameter determining the trade-off between the flatness of regression function f (i.e., $\frac{1}{2}\|w\|^2$) and the total tolerance on deviations larger than ϵ (i.e., $\sum_{i=1}^Q (\xi_i + \xi_i^*)$). A graphical description of ϵ -SVR model is shown in Figure 2. Points on the boundaries and outside the boundaries are called *Support Vectors*.

Figure 2. Graphic description of the ϵ -SVR model for a linear case.



To solve the optimization problem of Equation (8), a Lagrange function is constructed by introducing Lagrange multipliers a_i and a_i^* . Partial derivatives of the Lagrange function with respect to the primal variables (w, b, ξ_i, ξ_i^*) are calculated. Finally, the dual optimization problem can be derived by partial derivation of the Lagrange function and taking into account that $\xi_i \xi_i^* = 0$:

$$\text{maximize} \quad -\frac{1}{2} \sum_{i,j=1}^Q (a_i - a_i^*)(a_j - a_j^*)k(x_i x_j) - \epsilon \sum_{i=1}^Q (a_i + a_i^*) + \sum_{i=1}^Q y_i (a_i - a_i^*) \quad (9a)$$

$$\text{subject to} \quad \sum_{i=1}^Q (a_i - a_i^*) = 0 \quad \text{and} \quad a_i, a_i^* \in [0, C] \quad (9b)$$

Note that Support Vectors correspond to training data whose Lagrange multipliers a_i 's are non-zero in Equation (9). Training data for which $a_i = C$ are called Bounded Support Vectors, and they are located on the two boundaries in Figure 2. Training data with $0 < a_i < C$ are called Free Support Vectors and they are outside the boundaries [22]. Only Support Vectors contribute to the penalization in Equation (8a). Equation (9) can be solved by optimization algorithms such as Sequential Minimal Optimal, Chunking and Decomposing. The Sequential Minimal Optimal [23] is most frequently used

due to its fast calculation speed. Solving the quadratic problems of Equation (9) yields the final solution of ϵ -SVR:

$$f(x) = \sum_{i=1}^Q (a_i - a_i^*) k(x_i, x) + b \quad (10)$$

It is shown that ϵ -SVR has properties of robustness to noises and gross errors even though the gross errors are part of the set of Support Vectors [24,25]. This is because the Lagrange Multipliers solved in the objective function of Equation (9) are upper bounded by the constraint of Equation (9). All Support Vectors, including the gross errors, will have Lagrange Multipliers with absolute value no more than the upper bound C . From aspect of Equation (8), as C searches for trade-off between flatness and empirical risk, if necessary, ϵ -SVR will sacrifice the tolerance to get a good flatness. ϵ -SVR also has many other advantages such as global minima and reduced likelihood of overfitting.

2.3. Decoupling Process

The whole decoupling process consists of two stages. The first stage is for establishment of the coupling error functions. The second stage is in an actual on-line force perception task. During the task, the output voltages of all directions are obtained and the coupling error model is utilized to eliminate coupling errors and calculate measured forces. Our decoupling process is described as follows:

- (1) Conduct the static calibration experiment of a 3-axis force sensor and get the calibration data, which contain sets of input forces and output voltages ($f_{x(c)}$, $f_{y(c)}$, $f_{z(c)}$ and $u_{x(c)}$, $u_{y(c)}$, $u_{z(c)}$).
- (2) Do linear fitting of $f_{x(c)}$ and $u_{x(c)}$ from calibration data in X direction using LSM, to get the slope k_{xx} in Equation (2). In the same way, get coefficients k_{yy} and k_{zz} in Equation (2) from the calibration data in Y direction and Z direction respectively. In this way, three non-coupling functions $\omega_x()$, $\omega_y()$ and $\omega_z()$ and three inverse functions of the non-coupling functions $\omega_x^{-1}()$, $\omega_y^{-1}()$ and $\omega_z^{-1}()$ are obtained.
- (3) Do nonlinear function approximation for $u_{x(c)}$ and $u_{y(c)}$ from the calibration data in X direction by ϵ -SVR to get the corrected coupling function $\chi_{yx}()$, do nonlinear function approximation for $u_{x(c)}$ and $u_{z(c)}$ from the calibration data in X direction by ϵ -SVR to get the corrected coupling function $\chi_{zx}()$.
- (4) Calculate the corrected coupling functions $\chi_{xy}()$ and $\chi_{zy}()$ from calibration data in Y direction and $\chi_{xz}()$ and $\chi_{yz}()$ from calibration data in Z direction respectively in the same way as in step 3.

The above steps establish the entire coupling error model. The following steps are designed for decoupling based on the model using a 3-axis force sensor in an actual force perception task.

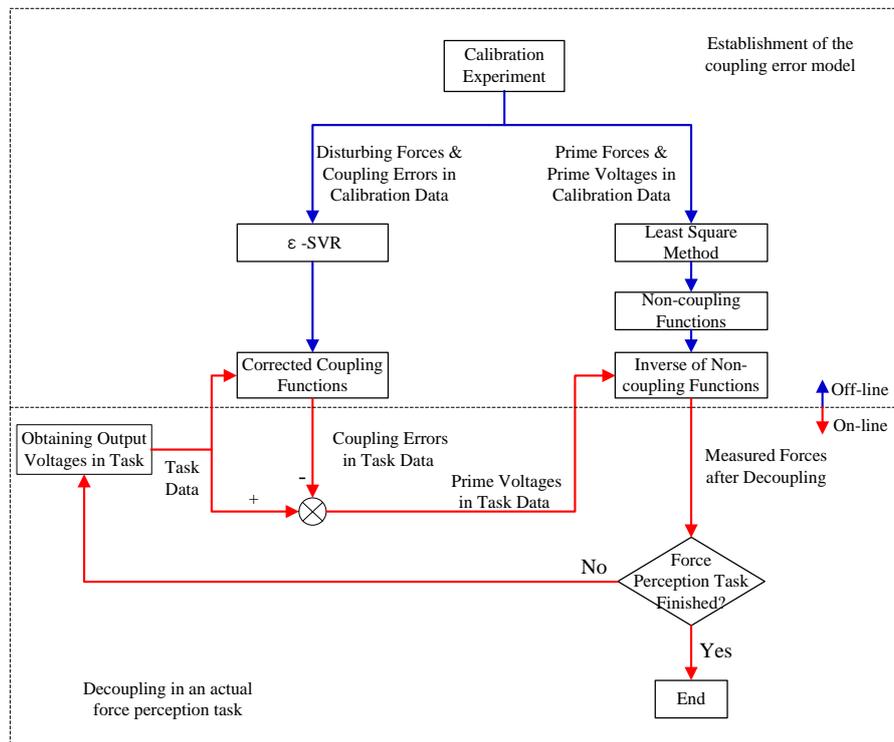
- (I) Obtain task data, namely, a set of output voltages ($u_{x(t)}$, $u_{y(t)}$, $u_{z(t)}$) of a 3-axis force sensor, during an actual force perception task. Compute each coupling error elements of measured data using the corrected coupling functions.

(II) Subtract the coupling errors from the output voltages to get prime voltages as shown in Equation (1). Then calculate prime forces of measured data, namely, the measured three dimensional forces by $\omega_x^{-1}()$, $\omega_y^{-1}()$ and $\omega_z^{-1}()$. The measured forces $f_{x(m)}, f_{y(m)}, f_{z(m)}$ can also be expressed in Equation (11).

$$\begin{cases} f_{x(m)} = w_x^{-1}[u_{x(t)} - \chi_{xy}(u_{y(t)}) - \chi_{xz}(u_{z(t)})] \\ f_{y(m)} = w_y^{-1}[u_{y(t)} - \chi_{yx}(u_{x(t)}) - \chi_{yz}(u_{z(t)})] \\ f_{z(m)} = w_z^{-1}[u_{z(t)} - \chi_{zx}(u_{x(t)}) - \chi_{zy}(u_{y(t)})] \end{cases} \quad (11)$$

Figure 3 shows the flow chart of the whole decoupling process. The blue arrows indicate off-line process of the establishment of the coupling error model with calibration data. The red arrows indicate on-line process of decoupling in an actual force perception task. Note that the coupling error model and decoupling process are proposed with respect to a predefined frame based on the structural characteristics of the sensor. During an actual force perception task, if the orientation of the predefined frame is different from that of the reference frame used for force perception, the measured forces after decoupling with respect to the predefined frame should be transformed by a rotation matrix in order to align with the reference frame.

Figure 3. The flow chart of the whole decoupling process.



3. Calibration Experiment

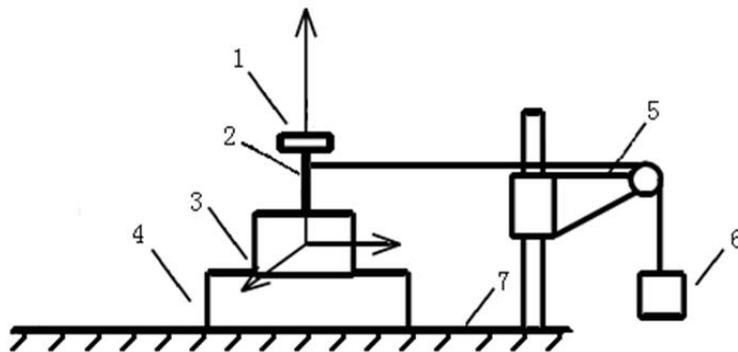
As an example 3-axis force sensor, we consider a 3-axis force sensor designed and fabricated in our lab. The force measurement range in both X direction and Y direction is -100 N to $+100\text{ N}$, and the force measurement range in Z direction is -150 N to 0 N . The corresponding output voltages range in

X direction and Y direction is -1.4 V to $+1.4\text{ V}$, and the output voltages range in Z direction is -1.2 V to $+1.2\text{ V}$.

The key component of the 3-axis force sensor is composed of a cross-beam elastic body with strain gauges, Wheatstone bridges, and a sampling circuit [26]. The principle of the 3-axis force sensor can be briefly described as follows: under the effect of loading forces, the cross-beam elastic body begins to deform measurably. Strain gauges that are pasted firmly on the surface of the cross-beam elastic body can sense the strains and transform them to changes in resistance. Resistance changes of strain gauges are measured by Wheatstone bridges and transformed to voltages. Finally, the sampling circuit amplifies the output voltages of Wheatstone bridges and transmits the amplified voltage signals to the robot controller.

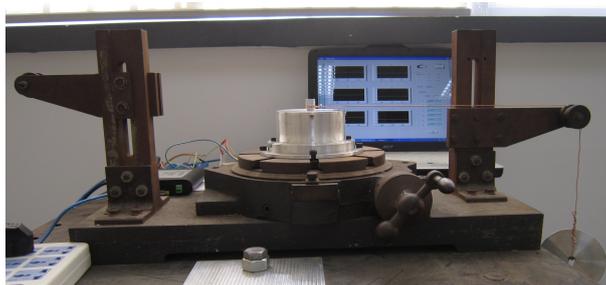
The common calibration method for medium range multi-axis force sensors [27–30] are utilized. A series of standard input forces in three directions are loaded on the sensor separately, and output voltages in all three directions are recorded simultaneously. The diagram of the static calibration experiment setup is shown in Figure 4, which consists of a loading plate, a calibration shaft, a 3-axis force sensor, an indexing plate, a pulley block, weights and a base.

Figure 4. Diagram of static calibration experiment setup: (1) loading plate, (2) calibration shaft, (3) three-axis force sensor, (4) indexing plate, (5) pulley block, (6) weights, (7) base.



The force sensor is fixed on a horizontally rotatable indexing plate with scale to ensure the direction of loading forces. The maximum error of the weights is $\pm 5\text{ mg}$. Loading forces in X direction and Y direction are generated by pulley block and weights while indexing plate is rotated by laboratory technicians to ensure the loading direction. During force loading process in Z direction, a series of standard weights are put directly on the loading plate. All loading forces are transferred from weights to force sensors through the calibration shaft. The experiment platform of static calibration experiment is shown in Figure 5.

The calibration range is from -100 N to $+100\text{ N}$ with an increment step of 10 N in X direction and Y direction. The calibration range in Z direction is from -150 N to 0 N with an increment step of 5 N . The whole calibration experiment is carefully repeated six times. Six sets of calibration data are obtained and zero-corrected. No gross errors are tested in the zero-corrected data.

Figure 5. The static calibration experiment setup.

4. Results

4.1. Decoupling by a Standard RBF

Cross validation is used to estimate the performance of decoupling by a standard MIMO RBF NN. The cross validation is repeated three times while each two sets of the zero-corrected calibration, named as “test data I” and “test data II”, are used for verification and the other four sets are used as training data. Matlab Neural Network Toolbox is used for simulations. The variance is set to 1, the maximum MSE error tolerance is set to 2.4×10^{-6} .

The maximum MSE error tolerance is chosen to be small enough to ensure that all training data be selected as centers of the MIMO RBF NN.

In order to test the decoupling accuracy, interference errors γ are calculated. Taking calibration data in X direction for example, the interference error γ_x is calculated in Equation (12),

$$\gamma_x = \frac{\max |f_{y(m)}| + \max |f_{z(m)}|}{f_{x(FS)}} \quad (12)$$

where $f_{x(FS)}$ denotes the full scale value of loaded force in X direction. The interference error γ_x accounts for maximum coupling errors in Y direction and Z direction induced by loaded force in X direction. γ_y and γ_z are calculated in the same way. Interference errors of initial data and data after decoupling by a standard MIMO RBF NN are shown in Table 1.

Table 1. Error analysis of initial data and decoupling by a standard MIMO RBF.

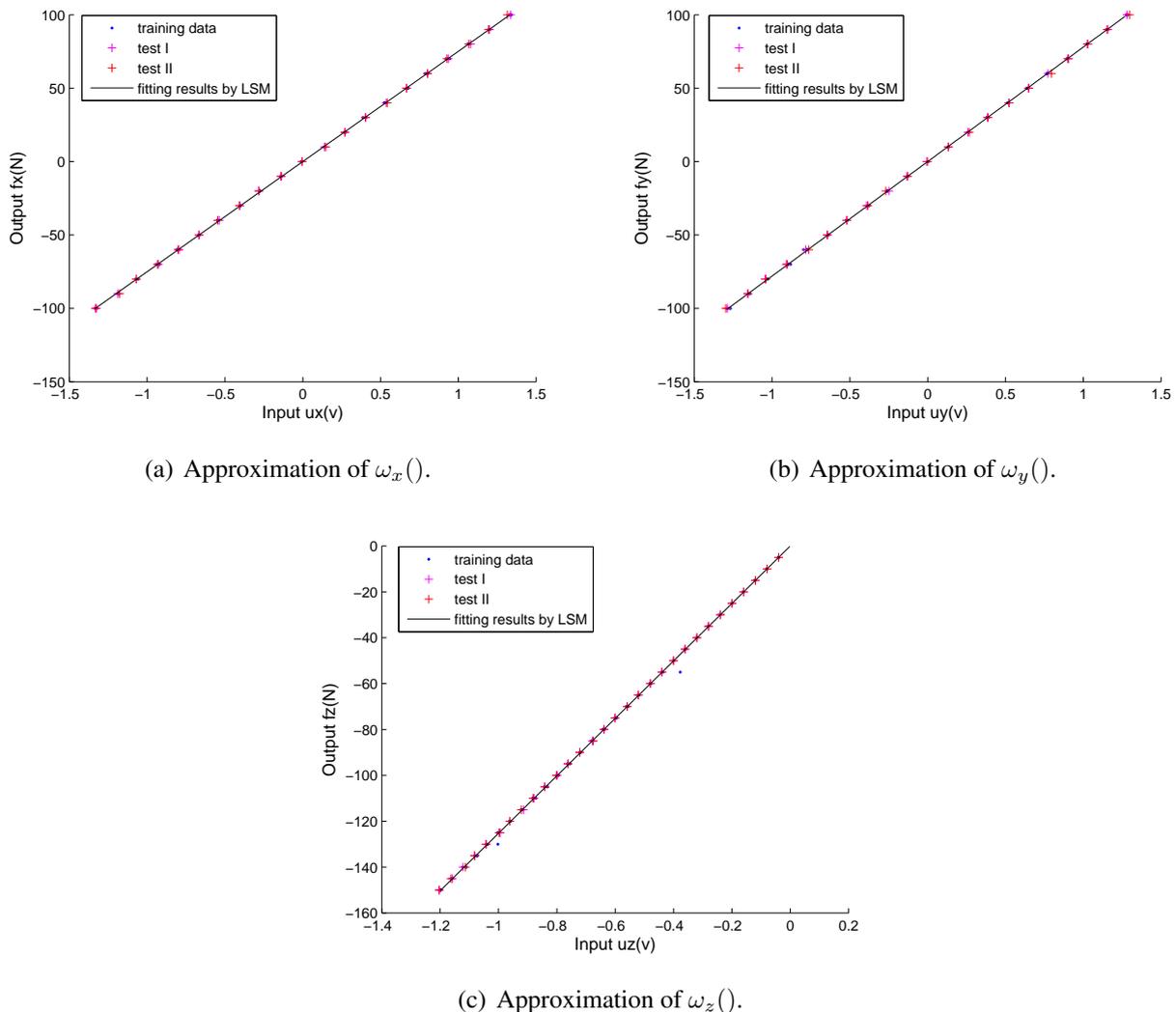
		γ_x		γ_y		γ_z	
		initial data	decoupled by RBF	initial data	decoupled by RBF	initial data	decoupled by RBF
the first	test I	2.54%	0.42%	2.57%	3.73%	2.36%	0.17%
cross-validation	test II	2.37%	0.61%	2.52%	7.99%	2.55%	0.23%
the second	test I	2.31%	1.02%	2.59%	2.32%	2.98%	0.71%
cross-validation	test II	2.73%	3.45%	2.65%	0.74%	2.81%	0.69%
the third	test I	2.40%	0.49%	2.68%	0.31%	2.57%	0.34%
cross-validation	test II	2.32%	0.26%	2.71%	0.31%	2.14%	0.32%

Table 1 shows that the decoupling algorithm by a standard MIMO RBF is unreliable, because although some interference errors are reduced to less than 1%, some interference errors are even worse than initial data (in boldface). This is because compared with the number of training data, there are too many parameters in the MIMO RBF NN. Also, different coupling error elements are not well related. For example, there is little relationship between e_{xy} and e_{xz} . However, decoupling by a standard MIMO RBF NN means all the relationships are regarded as the same in the training process. The standard MIMO RBF NN does not have a good conformability with the calibration data shape. As a result, random noise in the training data may easily contribute to overfitting.

4.2. Decoupling by the Coupling Error Model and ϵ -SVR

We also use cross validation to test the performance of our proposed decoupling algorithm. Here we take the first round of cross validation for example. To quantify the strong linearity of non-coupling functions, linear fittings are performed on the training data (plotted as blue points) to determine the slopes and strength of the correlations (plotted as black lines) as shown in Figure 6.

Figure 6. Linear fittings of non-coupling functions.



The linear correlation coefficients for all three directions are calculated to be more than 0.99, which correspond to assumption 1 (see Section 2). The inverse functions of non-coupling functions are obtained as:

$$\begin{cases} f_x = \omega_x^{-1}(u_x) = 75.065u_x \\ f_y = \omega_y^{-1}(u_y) = 77.788u_y \\ f_z = \omega_z^{-1}(u_z) = 125.17u_z \end{cases} \quad (13)$$

The six corrected coupling functions are approximated by ϵ -SVRs. Each corrected coupling function is fitted by a separate ϵ -SVR, where the pre-specified parameters are set as the same value. The ϵ -SVR based approximations of corrected coupling functions are carried out using MATLAB and LIBSVM [31] software applications. LIBSVM [31] is an integrated software for support vector classification, distribution estimation and regression. It implements an SMO-type optimization algorithm [32] and provides an efficient interface for MATLAB. The initial parameters used to train the ϵ -SVR model by LIBSVM tool are listed in Table 2.

Table 2. Initial parameters used by the LIBSVM train tool.

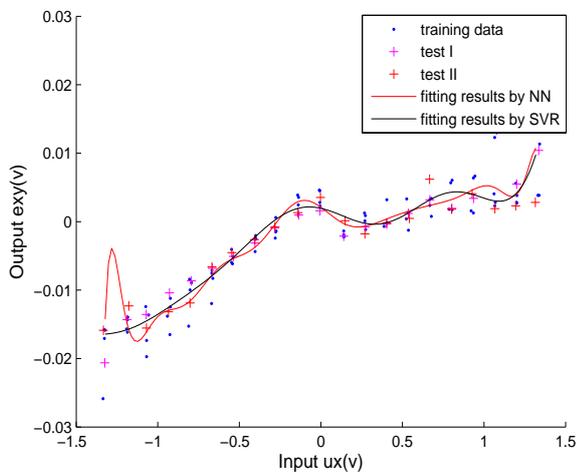
Parameter	Value
SVM type	3(ϵ -SVR)
Kernel function	2(RBF)
δ^2 (Standard deviation of kernel function)	1
C (Regularization parameter)	1000
ϵ (Deviation)	0.0001
Tolerance	0.001

Here we also employ six separate Single Input Single Output(SISO) RBF NNs to do nonlinear approximations of the corrected coupling error functions to make a comparison with ϵ -SVRs. Parameters of the SISO RBF NNs are defined the same as the MIMO RBF NN, which is mentioned in Section 4.1.

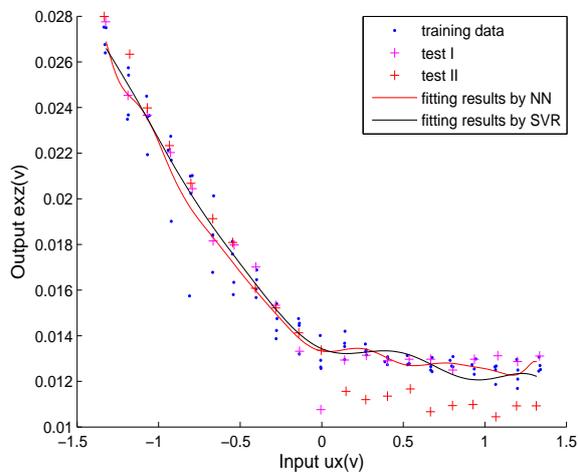
Figure 7 shows the fitting results of the six corrected coupling functions. The blue points represent training data, the red points and magenta points represent two groups of test data respectively, the black curves represent fitted corrected coupling functions using ϵ -SVR and the red curves represent fitted corrected coupling functions using SISO RBF NN. Relationships between disturbing forces and related coupling error elements are nonlinear and distinct from one another, which correspond to assumption 2 and assumption 3 (see Section 2.1). In Figures 7(a) and 7(d), obvious overfittings happen in the fitting results of χ_{xy} and χ_{yz} using SISO RBF NNs, but the ϵ -SVR provides a good generalization ability to approximate different kinds of nonlinear functions without overfitting.

In the same fashion, the other two rounds of cross validation are calculated and interference errors are listed in Table 3. In Table 3, all interference errors are reduced to less than 1.6% F.S. after decoupling by ϵ -SVR. Compared with Table 1, the decoupling accuracy is much more stable because no overfitting happens. Therefore, the proposed decoupling algorithm based on our coupling error model and ϵ -SVR is more robust.

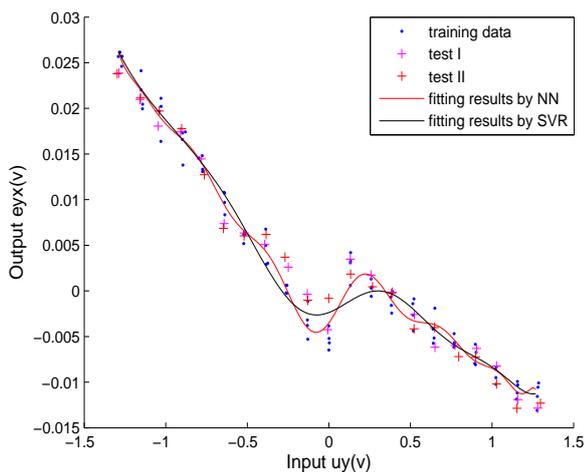
Figure 7. Approximation of corrected coupling functions by ϵ -SVR vs. by RBF NN.



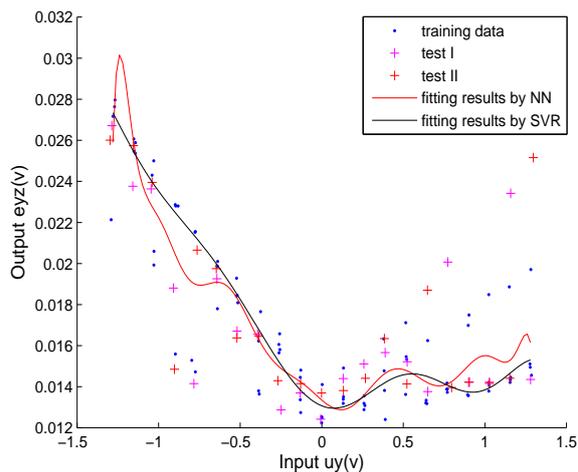
(a) Approximation of $\chi_{xy}(\cdot)$.



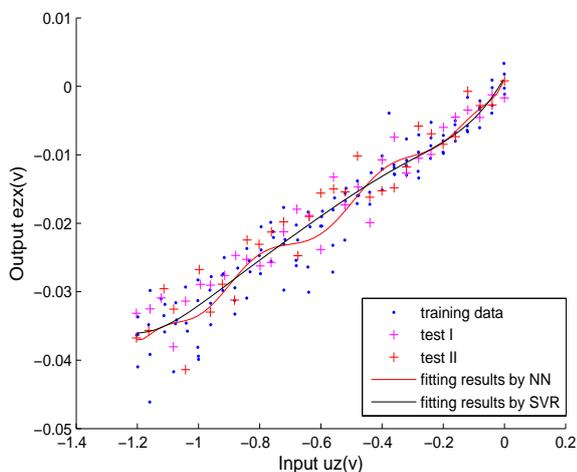
(b) Approximation of χ_{xz} .



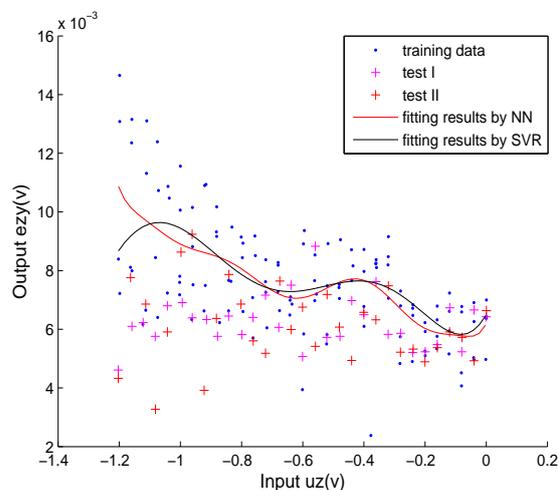
(c) Approximation of χ_{yx} .



(d) Approximation of χ_{yz} .



(e) Approximation of χ_{zx} .



(f) Approximation of χ_{zy} .

Table 3. Error analysis of decoupling algorithm by ϵ -SVRs.

		γ_x	γ_y	γ_z
the first	test I	1.69%	1.33%	0.82%
cross-validation	test II	1.45%	1.28%	0.59%
the second	test I	1.30%	1.47%	0.86%
cross-validation	test II	1.56%	1.24%	0.76%
the third	test I	1.43%	1.52%	0.67%
cross-validation	test II	1.48%	1.57%	0.83%

4.3. Processing Time

Besides decoupling accuracy, another crucial criterion for a successful decoupling algorithm is the processing speed of both model establishment and decoupling an output voltage vector in an actual force perception task, especially the latter for on-line task purposes. To evaluate the processing speed, the elapsed time of the above two stages is recorded and the whole process is repeated 10 times. All calculations are done on a Windows XP Inter(R) Xeon(TM) 2 QUAD CPU, 3.0 GHz processor with 3.00 GB RAM. An overview of the averaged elapsed time is shown in Table 4. Table 4 shows that the elapsed time of the proposed algorithm in both the model establishment and the decoupling of an output voltage vector is less than decoupling by a standard RBF NN. ϵ -SVR proves to be fast regressions. The elapsed time of decoupling a certain output voltage vector using a coupling error model based on ϵ -SVR is less than 0.001 s. This means that the decoupling frequency is higher than 1,000 Hz. The speed of testing process based on ϵ -SVR is fast.

Table 4. Mean elapsed time (in seconds) of the training process and the testing process.

	a RBF NN	ϵ -SVRs
Establishment of the model	80.5603	3.3702
Decoupling of an output voltage vector	0.1404	0.00093

Summarizing, the calculation results of the decoupling methods demonstrate that our decoupling method based on our coupling error model and ϵ -SVRs has high reliability and fast running speed when no gross errors exist in the calibration data. In the next section, we will artificially add gross errors to the calibration data and analyze the decoupling method's robustness to gross errors.

4.4. Robustness to Gross Errors

Now we simulate the decoupling algorithm's reaction to gross errors. The gross errors in calibration data are also called *outliers*. As the quantity of gross errors in calibration data is always small, in our research, only one gross error is generated in one dimension while the calibration data of other dimensions remain the same. Here we take the first round of cross validation for example. The decoupling process is the same as Section 4.2. We artificially generate a gross error into the calibration

data in X direction to test the gross error’s effect on the decoupling accuracy in all dimensions. The gross error is added artificially by arbitrarily moving a point away from its initial location.

As no gross errors are introduced into the calibration data of other dimensions, the fitting results of functions in Y direction and Z direction will remain the same as shown in Figures 6 and 7. The fitting results of the non-coupling function $\omega_x()$ and corrected-coupling functions $\chi_{xy}()$ and $\chi_{xz}()$ in X direction with the gross error (plotted with red circles) are shown in Figures 8 and 9. The inverse function of $\omega_x()$ is obtained as:

$$f_x = \omega_x^{-1}(u_x) = 74.193u_x \tag{14}$$

Figure 8. The fitting result of the ω_x for calibration data with gross error.

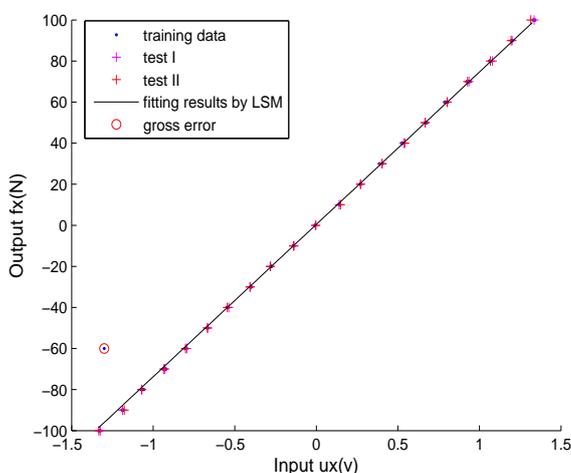
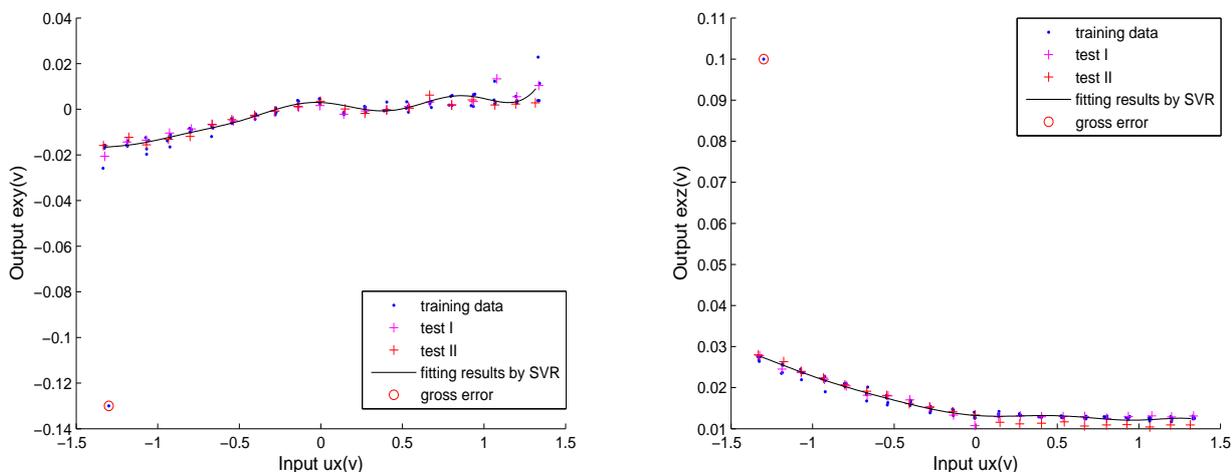


Figure 9. The fitting results of $\chi_{xy}()$ and χ_{xz} for calibration data with the gross error.



(a) Approximation of $\chi_{xy}()$.

(b) Approximation of χ_{xz} .

Comparing Figure 9 with Figure 7, when the obtained calibration data contain a gross error (outlier), the ϵ -SVR is not sensitive to the gross error and performs with high reliability, which correspond to the

analysis in Section 2.2.2. Consequently, the proposed decoupling algorithm based on ϵ -SVR displays a strong robustness to the gross error. The calculated interference errors of test data I and test data II under the gross error's effect are listed in Table 5. Comparing Table 5 with Table 3, when the gross error is artificially introduced, γ_x , γ_y and γ_z after decoupling by ϵ -SVR of both test data I and test data II seem almost unchanged. The comparison of interference errors also confirms that ϵ -SVR is robust to gross errors.

Table 5. Error analysis of decoupling by ϵ -SVRs when a gross error is introduced.

		γ_x	γ_y	γ_z
the first	test I	1.41%	1.51%	0.71%
cross-validation	test II	1.44%	1.56%	0.87%

5. Conclusions

In this paper, a robust decoupling algorithm to efficiently reduce coupling errors for 3-axis force sensors is presented. The decoupling algorithm is based on the establishment of a novel coupling error model and ϵ -SVR. In the coupling error model, input forces are partitioned into prime forces and disturbing forces, and the corresponding output voltages are partitioned into prime voltages and coupling error elements. The structure of the coupling error model makes the decoupling process in one dimension separated from other dimensions. Linear relationships between prime forces and prime voltages in every dimension are fitted by LSM. Nonlinear relationships between disturbing forces and corresponding coupling error elements are fitted by ϵ -SVR.

The experimental results show the effectiveness of the decoupling method. All interference errors are reduced to less than 1.6%. Compared with decoupling algorithm using one standard MIMO RBF NN, the proposed decoupling algorithm demonstrates a better tolerance to noises and faster calculation speed (with the decoupling frequency higher than 1,000 Hz). It also shows robustness to gross errors. As a result, compared with existing decoupling methods, the decoupling algorithm based on the proposed coupling error model and ϵ -SVR is much more reliable in complex scenarios.

Acknowledgments

This work was supported in part by the National Natural High Technology Research and Development Program of China (863 Program) (No. 2008AA040202), Natural Science Foundation Project of Jiangsu Province (No. DK2010063), Scientific and Technological Research Project of Changzhou (No. CE20100022).

References

1. Perry, D. Multi-axis force and torque sensing. *Sens. Rev.* **1997**, *17*, 117–120.
2. Lefebvre, T.; Xiao, J.; Bruyninckx, H.; De Gerssem, G. Active compliant motion: A survey. *Adv. Robot.* **2005**, *19*, 479–499.

3. Siciliano, B.; Villani, L. *Robot Force Control*; Springer: Napoli, Italy, 1999; Volume 540.
4. Siciliano, B.; Khatib, O., Force Control. In *Springer Handbook of Robotics*; Springer-Verlag New York Inc: New York, NY, USA, 2008; pp. 161–185.
5. Xu, G.; Song, A.; Li, H. Adaptive impedance control for upper-limb rehabilitation robot using evolutionary dynamic recurrent fuzzy neural network. *J. Intell. Robot. Syst.* **2011**, *62*, 501–525.
6. Nagata, F.; Kusumoto, Y.; Watanabe, K.; Tsuda, K.; Yasuda, K.; Yokoyama, K.; Omoto, M.; Miyako, H. Polishing Robot for PET Bottle Molds Using a Learning-Based Hybrid Position/Force Controller. In *Proceedings of the 5th Asian Control Conference*, Melbourne, Australia, June 2004; pp. 914–921.
7. Lee, S.; Asada, H. A perturbation/correlation method for force guided robot assembly. *IEEE Trans. Robot. Autom.* **1999**, *15*, 764–773.
8. Song, A.; Wu, J.; Qin, G.; Huang, W. A novel self-decoupled four degree-of-freedom wrist force/torque sensor. *Measurement* **2007**, *40*, 883–891.
9. Beyeler, F.; Muntwyler, S.; Nelson, B. Design and Calibration of a Microfabricated 6-Axis Force-Torque Sensor for Microrobotic Applications. In *Proceedings of the International Conference on Robotics and Automation*, Zurich, Switzerland, May 2009; pp. 520–525.
10. Kim, K.; Sun, Y.; Voyles, R.; Nelson, B. Calibration of multi-axis MEMS force sensors using the shape-from-motion method. *IEEE Sens. J.* **2007**, *7*, 344–351.
11. Oddo, C.; Valdastrì, P.; Beccai, L.; Roccella, S.; Carrozza, M.; Dario, P. Investigation on calibration methods for multi-axis, linear and redundant force sensors. *Meas. Sci. Technol.* **2007**, *18*, 623–631.
12. Voyles, R.; Morrow, J.; Khosla, P. Shape from Motion Approach to Rapid and Precise Force/Torque Sensor Calibration. In *Proceedings of the International Mechanical Engineering Congress and Exposition*, San Francisco, USA, November 1995; pp. 67–73.
13. Cao, H.; Yu, Y.; Ge, Y. A Research of Multi-Axis Force Sensor Static Decoupling Method Based on Neural Network. In *Proceedings of the International Conference on Automation and Logistics*, Shenyang, China, August 2009; pp. 875–879.
14. Lei, J.; Qiu, L.; Liu, M.; Song, Q.; Ge, Y. Application of Neural Network to Nonlinear Static Decoupling of Robot Wrist Force Sensor. In *Proceedings of the 6th World Congress on Intelligent Control and Automation*, Dalian, China, October 2006; pp. 5282–5285.
15. Ming, D.; Zhang, X.; Liu, X.; Wan, B.; Hu, Y.; Luk, K. Nonlinear Static Decoupling of Six-Dimension Force Sensor for Walker Dynamometer System Based on Artificial Neural Network. In *Proceedings of the International Conference on Computational Intelligence for Measurement Systems and Applications*, Hong Kong, May 2009; pp. 14–17.
16. Cristianini, N.; Shawe-Taylor, J. *An Introduction to Support Vector Machines: And Other Kernel-Based Learning Methods*; Cambridge Univ. Press: Cambridge, UK, 2000.
17. Guo, J.; He, Y.; Liu, C. Nonlinear correction of photoelectric displacement sensor based on least square support vector machine. *J. Cent. S. Univ. Technol.* **2011**, *18*, 1614–1618.
18. Wang, X. Non-Linearity Estimation and Temperature Compensation of Capacitor Pressure Sensors Using Least Square Support Vector Regression. In *Proceedings of the International Symposium on Knowledge Acquisition and Modeling Workshop*, Wuhan, China, December 2008; pp. 1161–1164.
19. Smola, A.; Schölkopf, B. A tutorial on support vector regression. *Stat. Comput.* **2004**, *14*, 199–222.

20. Vapnik, V. *Statistical Learning Theory*; Wiley: New York, NY, USA, 1998.
21. Verrelst, J.; Muñoz, J.; Alonso, L.; Delegido, J.; Rivera, J.; Camps-Valls, G.; Moreno, J. Machine learning regression algorithms for biophysical parameter retrieval: Opportunities for sentinel-2 and-3. *Remote Sens. Environ.* **2012**, *118*, 127–139.
22. Wang, L. *Support Vector Machines: Theory and Applications*; Springer Verlag: Singapore, 2005; Volume 177.
23. Platt, J.C. Fast Training of Support Vector Machines Using Sequential Minimal Optimization. In *Advances in Kernel Methods: Support Vector Machines*; MIT Press: Cambridge, MA, USA, 1998; pp. 185–208.
24. Dahai, L.; Tianshi, L. On-Line Robust Modeling of Nonlinear Systems Using Support Vector Regression. In *Proceedings of the 2nd International Conference on Advanced Computer Control*, Shenyang, China, March 2010; pp. 204–208.
25. Chuang, C.; Su, S.; Jeng, J.; Hsiao, C. Robust support vector regression networks for function approximation with outliers. *IEEE Trans. Neural Netw.* **2002**, *13*, 1322–1330.
26. Ma, J.; Song, A. Development of a novel two-axis force sensor for chinese massage robot. *Appl. Mech. Mater.* **2012**, *103*, 299–304.
27. Berkelman, P.; Whitcomb, L.; Taylor, R.; Jensen, P. A miniature microsurgical instrument tip force sensor for enhanced force feedback during robot-assisted manipulation. *IEEE Trans. Robot. Autom.* **2003**, *19*, 917–921.
28. Chao, L.; Yin, C. The six-component force sensor for measuring the loading of the feet in locomotion. *Mater. Des.* **1999**, *20*, 237–244.
29. Chen, L.; Song, A. A Novel Three Degree-of-Freedom Force Sensor. In *Proceedings of the International Conference on Measuring Technology and Mechatronics Automation*, Zhangjiajie, China, April 2009; pp. 77–80.
30. Xu, K.; Li, C.; Zhu, Z. Dynamic modeling and compensation of robot six-axis wrist force/torque sensor. *IEEE Trans. Instrum. Meas.* **2007**, *56*, 2094–2100.
31. Chang, C.C.; Lin, C.J. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.* **2011**, *2*, 1–27.
32. Fan, R.; Chen, P.; Lin, C. Working set selection using second order information for training support vector machines. *J. Mach. Learn. Res.* **2005**, *6*, 1889–1918.