LEFT-INVERSE SYSTEM DYNAMIC DECOUPLING AND COMPENSATING METHOD USING NEURAL NETWORKS

Dongchuan Yu, Aiguo Wu

School of Electrical Engineering and Automation,
Tianjin University, Tianjin 300072 P.R. China

Abstract: A novel and practical *left-inverse system based neural networks dynamic decoupling and compensating* (*LISNNDDC*) method is proposed for improving dynamic performance of generic nonlinear muti-dimension sensors (e.g. muti-axis force sensors) instead of well-used ones. Consequently, the proposed method is not only of prime theoretical interest but also, in practical implementation, can obtain better dynamic performance. A six-axis wrist force sensor is illustrated as an example to validate that the proposed method can markedly improve dynamic performance of the muti-dimension sensors and is superior to previous methods. *Copyright © 2005 IFAC*

Keywords: Dynamic decoupling, dynamic compensation, inverse system, multidimensional sensor, neural networks.

1. INTRODUCTION

Recently, multi-axis force/moment sensors, which can measure multi-axis force and moment, are widely implemented for robotic systems and also for wind-tunnel balances, automobiles and shipbuilding (Liu and Tzo, 2002; Perry, 1997; Chao and Chen, 1997; Dubious, 1981). However, these sensors cannot be used for the case of that high dynamic measurement accuracy is required, such as robot operations e.g. assembly, welding and grinding. There are two key reasons described as follows (Xu and Li, 2000).

1). Damped ratio of the multi-axis force/moment sensors is small and their natural frequency is low. As a result, dynamic response of the sensors is slow and the settling time is long.

2). The elastic body of the sensors is an integer structure and the interactions of various channels exist and cannot be avoided completely. To make matter worse, the dynamic characteristics of those interactions are markedly nonlinear.

Therefore, nonlinear dynamic coupling and slow dynamic response are two main problems influencing on the dynamic performance of the muti-axis force/moment sensors. An available and economical technique to solve the two problems is so-called *dynamic decoupling and compensating* (DDC) method, which uses certain algorithm to decouple dynamic interactions between axes and meanwhile, improve the dynamic performance of every channel. Some DDC methods based on linear system theory have been introduced and implemented to improve the dynamic performance of the muti-axis force sensors (Xu and Li, 2000). However, since the dynamic characteristic of the decoupling between axes is markedly nonlinear. Obviously, those methods cannot simultaneously and thoroughly solve the two problems.

In the paper, a novel and practical *left-inverse system based neural networks dynamic decoupling and compensating* (*LISNNDDC*) method is proposed and can be used for generic nonlinear muti-dimension sensors e.g. multi-axis force/moment sensors. It is not only of prime theoretical interest but also for practical implementation purposes to obtain better dynamic performance.

The rest paper is planned as follows. The principle of left-inverse DDC, which is the basis of the proposed *LISNNDDC* method, will be introduced in Section 2, then in Section 3, *LISNNDDC* method and its implement steps are clearly explained therein. In section 4, a six-axis wrist force sensor is illustrated as an example to validate that the *LISNNDDC* method can markedly improve dynamic performance of muti-dimension sensors.

2. PRINCIPLE OF LEFT-INVERSE SYSTEM DDC

Definition 1 (*left-inverse system definition*): Let one system $\Sigma : \vec{\mathbf{u}} \to \vec{\mathbf{y}}$. If there exists an according system $\prod \ \vec{\mathbf{v}} \to \vec{\mathbf{w}}$ such that $\vec{\mathbf{w}}(t) = \vec{\mathbf{u}}(t)$ if $\vec{\mathbf{v}}(t) = \vec{\mathbf{y}}(t)$, then the original system Σ is left-invertible and the system $\prod$ is called left-inverse system of the original system Σ , where

$$\vec{\mathbf{u}} = [u_1, \cdots, u_p]^T, \vec{\mathbf{y}} = [y_1, \cdots, y_p]^T, \vec{\mathbf{v}} = [v_1, \cdots, v_p]^T, \vec{\mathbf{w}} = [w_1, \cdots, w_p]^T.$$

From the Definition 1, it is easily known that, if the outputs of the original system Σ are fed to the left-inverse system $\prod$, the composite operator $\prod \circ \Sigma$ is a unit operator. Namely, the outputs of left-inverse system $\prod$ can reconstruct the inputs of original system Σ . As a result, if a muti-dimension sensor is considered as a dynamic system and its left-inverse system exists, the left-inverse system realizes an idea DDC of the muti-dimension sensor. We call the method *left-inverse system DDC* (LISDDC) in the paper and its principle is shown in Fig. 1.

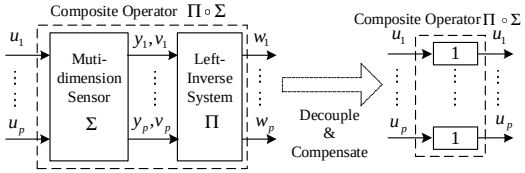

Fig. 1. The principle of the LISDDC for muti-dimension sensor.

Consider a muti-dimension sensor Σ

$$\mathbf{F}(\mathbf{Y}^{(\mathbf{N})}, \mathbf{Y}, \mathbf{U}^{(\mathbf{M})}, \mathbf{U}) = \mathbf{0} \tag{1}$$

where $\mathbf{F}(\cdot)$ is locally analytic function and

$$\begin{aligned}
&\mathbf{N} = (n_1, n_2, \cdots, n_p), \mathbf{M} = (m_1, m_2, \cdots, m_p), \\
&\mathbf{Y} = [y_1, y_2, \cdots, y_p], \mathbf{Y}^{(\mathbf{N})} = [Y_1, Y_2, \cdots, Y_p]^T \\
&\mathbf{Y}_i = [\dot{y}_i, y_i^{(2)}, \cdots, y_i^{(n_i)}]^T, \mathbf{U} = [u_1, u_2, \cdots, u_p]^T, \\
&\mathbf{U}^{(\mathbf{M})} = [U_1, U_2, \cdots, U_p]^T, \\
&\mathbf{U}_i = [\dot{u}_i, u_i^{(2)}, \cdots, u_i^{(m_i)}]^T, i \in \{1,2, \cdots, p\} \\
&\mathbf{F}(\cdot) = [F_1(\cdot), F_2(\cdot), \cdots, F_p(\cdot)]^T.
\end{aligned} \tag{2}$$

It is obviously, (1) is a very generic form and can describe most muti-dimension sensors.

The next Theorem 1 can be easily derived from implicit function theorem.

Theorem 1: For muti-dimension sensor system Σ described as (1), if $\partial F / \partial U$ is non-singular and everywhere continuous on certain open set D, the system Σ is invertible on D and

$$\mathbf{U} = \mathbf{F}^{-1}(\mathbf{Y}^{(\mathbf{N})}, \mathbf{Y}, \mathbf{U}^{(\mathbf{M})}) . \tag{3}$$

Remark 1: Actually, the Theorem 1 gives the existence condition of inverse system and its analytic expression (3) which is the LISDDCor for muti-dimension sensor system Σ described as (1). □□

However, there are some problems in realizing the LISDDCor in engineering practice. Firstly, the invertiblility of sensor system cannot be easily judged if the dynamic characteristic of the sensor is inaccurate. In addition, even though the dynamic characteristic of the sensor can be accurately known, analytic expression of the inverse system is not easily achieved if the dynamic characteristic of the sensor is markedly nonlinear. Thirdly, since differential is sensitive to noise and external disturbances, how to get robust and correct high-order differential is still a long-term problem in engineering practice.

Trying to solve those problems, a novel *LISNNDDC* method is proposed, where both an approximate differential algorithm and neural networks are incorporated with LISDDC method. The forthcoming analysis results will clearly show that the *LISNNDDC* method is feasible and can be implemented in engineering practice.

3. LISNNDDC METHOD

3.1 Approximate differential algorithm

Using the definition of differential yields

$$\dot{x}(t) = \lim_{d \to 0} \frac{x(t) - x(t-d)}{d} \tag{4}$$

where d>0.

Obviously, (4) can be re-described as follows.

For arbitrary positive number ε , there exists δ dependent on ε , such that $\forall d < \delta$

$$\left| \dot{x}(t) - \frac{x(t) - x(t-d)}{d} \right| < \varepsilon. \tag{5}$$

Consequently, for arbitrary time t, $\dot{x}(t)$ can be approximated with arbitrary accuracy using two sufficiently close sequent points, i.e.,

$$\dot{x}(t) \approx h_1[d, x(t), x(t-d)]. \tag{6}$$

Using the definition of differential yields

$$\begin{aligned} \ddot{x}(t) &= \lim_{d\to 0}\frac{\dot{x}(t)-\dot{x}(t-d)}{d} \\ &= \lim_{d\to 0}\frac{1}{d}[\lim_{d\to 0}\frac{x(t)-x(t-d)}{d} - \lim_{d\to 0}\frac{x(t-d)-x(t-2d)}{d}]. \end{aligned} \tag{7}$$

Similarly, that means

$$\ddot{x}(t) \approx h_2[d, x(t), x(t-d), x(t-2d)]. \tag{8}$$

By induction, the i th order differential can be approximated as

$$x^{(i)}(t) \approx h_i[d, x(t), x(t-d), \cdots x(t-id)]. \tag{9}$$

The creditability of (9) can also be validated in the well- known finite difference theory (Jordan, 1960).

3.2 Function approximation algorithm via approximate differential

From (9) and Taylor's formula, the next Theorem 2 can be easily derived.

Theorem 2: For any $\varepsilon > 0$, there exists δ independent on ε such that $\forall d < \delta$

$$| f(x, \dot{x}, \cdots, x^{(i)}) - \bar{f}[x(t), x(t-d), \cdots x(t-id)]| < \varepsilon \tag{10}$$

where $\bar{f}[x(t), \cdots x(t-id)] = f[x(t), h_1, \cdots h_i]$ and $h_j = h_j[d, x(t), x(t-d), \cdots x(t-jd)],\ j \in \{1,2,\cdots,i\}$.

Based on Theorem 2, for any $\varepsilon > 0$, $\exists \delta(\varepsilon)$ and $\boldsymbol{\psi}$, such that $\forall d < \delta(\varepsilon)$

$$\mathbf{F}^{-1}(\mathbf{Y}^{(\mathbf{N})}, \mathbf{Y}, \mathbf{U}^{(\mathbf{M})}) = \boldsymbol{\psi}(\mathbf{X}, \mathbf{U}) + \boldsymbol{\gamma}(\mathbf{X}, \mathbf{U}) \tag{11}$$

where $\| \boldsymbol{\gamma}(\mathbf{X}, \mathbf{U}) \| < \varepsilon$ and

$$\begin{aligned} &\mathbf{X} = [\mathbf{1}, \mathbf{Y}^{\mathbf{T}}, (\mathbf{Y_d})^{\mathbf{T}}, (\mathbf{U_d})^{\mathbf{T}}]^{\mathbf{T}}, \mathbf{Y} = [y_1, y_2, \cdots, y_p]^T, \\ &\mathbf{Y}_d = [\mathbf{Y}_{d1}, \mathbf{Y}_{d2}, \cdots, \mathbf{Y}_{dp}]^T, \mathbf{Y}_{di} = [y_i(t-d), \cdots, y_i(t-\bar{n}_i d)]^T, \\ &\mathbf{U} = [u_1, u_2, \cdots, u_p]^T, \mathbf{U}_d = [\mathbf{U}_{d1}, \mathbf{U}_{d2}, \cdots, \mathbf{U}_{dp}]^T, \\ &\mathbf{U}_{di} = [u_i(t-d), \cdots, u_i(t-\bar{m}_i d)]^T, \bar{n}_i \geq n_i, \bar{m}_i \geq m_i, \\ &i \in \{1,2,\cdots,p\}. \end{aligned} \tag{12}$$

Substituting (3) into (11) produces

$$\mathbf{U} = \boldsymbol{\psi}(\mathbf{X}, \mathbf{U}) + \boldsymbol{\gamma}(\mathbf{X}, \mathbf{U}). \tag{13}$$

Obviously, for (13), both left and right side include $\mathbf{U}$. To guarantee the existence of the solution of (13), the next assumption is needed.

Assumption 1: $\dfrac{\partial \boldsymbol{\psi}}{\partial \mathbf{U}} + \dfrac{\partial \boldsymbol{\gamma}}{\partial \mathbf{U}} \neq 1$.

Based on Assumption 1 and implicit function theorem, for (13), we can have

$$\mathbf{U} = \overline{\boldsymbol{\psi}}(\mathbf{X}). \tag{14}$$

3.3 Neural networks for function approximation

It is well known that neural networks can approximate nonlinear function with arbitrary accuracy if the architecture and training algorithm of neural networks are appropriately chosen. Without loss of generality, the backpropagation (BP) neural networks are singled out in the paper. The next Assumption 2 is credible.

Assumption 2: For any $\varepsilon > 0$, there exist W, V such that

$$\left\| \overline{\boldsymbol{\psi}}(\mathbf{X}) - \mathbf{W}^T \sigma(\mathbf{V}^T \mathbf{X}) \right\| < \varepsilon \tag{15}$$

where $\sigma(\cdot)$ is a sigmoid function defined as $\sigma(z) = 1/(1+e^{\alpha z})$ and $\mathbf{X}$ is defined in (12).

3.4 LISNNDDCor of muti-dimension sensor

By substituting (14) into (15), we know

$$\left\| \mathbf{U} - \mathbf{W}^T \sigma(\mathbf{V}^T \mathbf{X}) \right\| < \varepsilon. \tag{16}$$

That implies

$$\mathbf{U} \approx \mathbf{W}^T \sigma(\mathbf{V}^T \mathbf{X}). \tag{17}$$

Consequently, (17) actually acts as the *LISNNDDCor*. Training the BP neural networks in *LISNNDDCor* plotted in Fig 2 can realize DDC of muti-dimension sensors as shown in Fig. 3.

Remark 2: The differential of signals has been incorporated with neural networks and cannot directly be discerned from the *LISNNDDC* method described in (17), although the approximate differential algorithm using finite difference theory is introduced. Then, it is powerful and practical and can be used for noise-corrupted measurement data.□□

Remark 3: In practice, we assume that left-inverse system of the sensor exists and both Assumption 1 and Assumption 2 are appropriate and correct. Those

assumptions don't need to be validated one by one. Obviously, we can believe that those assumptions are true if the experiment results of DDC are perfect as desired. Namely, in practical implement of the *LISNNDDCor*, we can directly use the result (17) and don't need to validate those assumptions true.□□

Remark 4: The practical implement of *LISNNDDCor* can be broken into five steps described as follows.

Step 1: For knowing the dynamic performance of a sensor and getting data for training neural networks, dynamic calibrating experiments need to be done.

Step 2: Chose the order of the sensor, i.e., $\mathbf{N}=(n_1,n_2,\cdots,n_p)$, $\mathbf{M}=(m_1,m_2,\cdots,m_p)$.

Step 3: Chose the delay time d, which cannot be overlarge, neither can be oversmall. In practice, $1/d$ can be chosen between quintupling and decuple of maximum oscillation frequency of the muti-dimension sensor.

Step 4: Train the BP neural networks. Before training the neural networks, neural networks architecture, training algorithm and training num need to be chosen. In addition, the data of dynamic experiments should be pre-processed for eliminating or reducing noise and external disturbances, and also for normalization.

Step 5: Implement the *LISNNDDCor* and validate if the dynamic performance can satisfy the desired accuracy. If it is true, we accomplish the DDC of the sensor; if not, we must skip back to the Step 2 and re-choose parameters till the desired accuracy is satisfied.□□

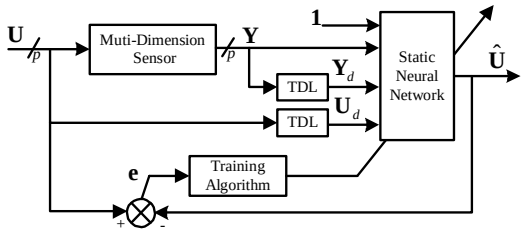

Fig.2. The block diagram of neural network training with the tapped-delay-lines (TDL) defined in (12).

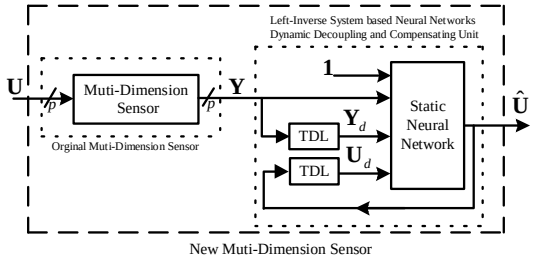

Fig.3. The block diagram of the *LISNNDDCor* with the tapped-delay-lines (TDL) defined in (12).

4. LISNNDDC EXPERIMENTS FOR A SIX-AXIS WRIST FORCE SENSOR

4.1 Estimation rules for the effect of DDC

The next two rules are introduced to estimate the effect of DDC and can also be found in (Xu and Li, 2000).

(1). The standard derivation

$$\delta=\sqrt{\frac{(\Omega_i-\Gamma_i)^2}{L-1}} \tag{18}$$

where Ω_i denotes input signals of the sensor and Γ_i consequent decoupled and compensated output signals; L denotes total number of the data.

(2). The relative error

$$E_r=\frac{(\Omega_i-\Gamma_i)_{\max}}{|\Omega_i|_{\max}}\times 100\%. \tag{19}$$

Obviously, the smaller the values of δ, E_r, the better dynamic decoupling and compensating results and the higher dynamic-measurement accuracy of the sensor.

4.2 Dynamic calibrating experiments of a six-axis wrist force sensor

A six-axis wrist force sensor is chosen to validate the efficiency of the proposed *LISNNDDC* method. To know the dynamic performance of the chosen six-axis wrist force sensor and obtain the data for training neural networks, dynamic calibrating experiments of the sensor should be accomplished using the dynamic calibrating system plotted in Fig.4.

Using frequency analyzer knows the maximum oscillation frequency of the six-axis wrist force sensor is about 1kHz and the frequency of the impulse force generated by a hammer ranges between 0 and 2kHz. Therefore, the hammer can be considered as a practical impulse force generator. In the head of the hammer, a piezoelectric sensor is installed to transform the impulse force into an electric charge signal, which is amplified by a charge amplifier and sent to a computer data receiver system. Meanwhile, a transient recorder is used for recording the response of the six-axis wrist force sensor when impulse force is applied to the sensor using the hammer and consequent response data will also be sent to the *computer data receiver system.*

The six-axis wrist force sensor is fixed on a testing platform for dynamic calibration and is knocked by the hammer respectively along X, Y and Z directions. Many experiments have clearly shown that the decoupling between axes of the chosen sensor is

markedly nonlinear. Namely, when certain direction of the sensor is knocked, other directions outputs will show nonlinear dynamic characteristic instead of zero. Some experiment results of decoupling between Z and Y directions of the chosen six-axis wrist force sensor are plotted in Fig. 5.

From Fig. 5, it is easily found that, when impulse force is imported from Z/Y direction, consequent response of Y/Z direction is markedly nonlinear rather than zero. That means that the decoupling between Y and Z direction is serious and markedly nonlinear.

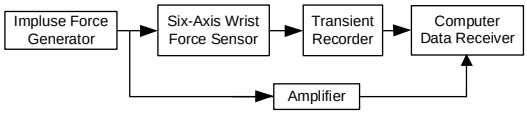

Fig. 4. Dynamic calibrating system for six-axis wrist force sensor.

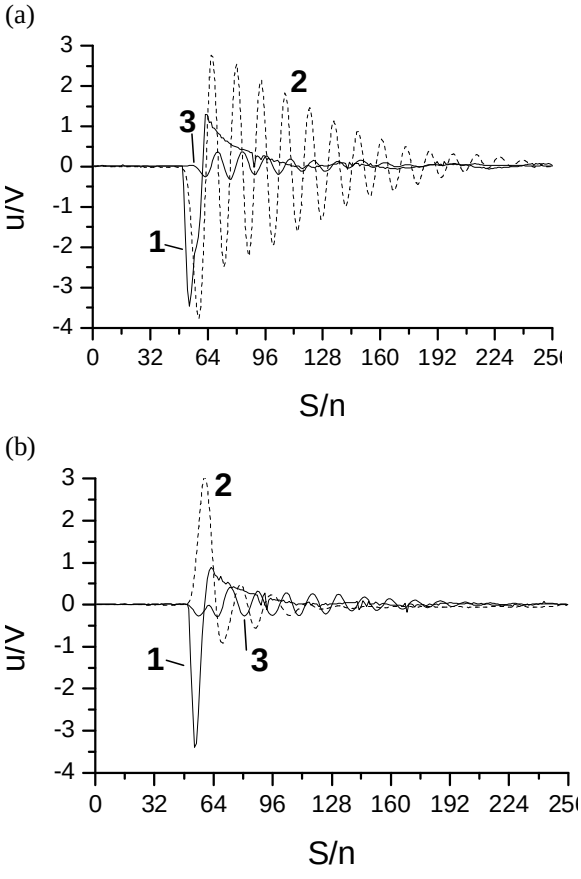

Fig.5. Experiment results of decoupling between Y and Z directions for (a) impulse force imported from Z direction and consequent responses with: 1-impulse force, 2-response of Z direction, 3-response of Y direction; and (b) impulse force imported from Y direction and consequent responses with: 1-impluse force, 2-response of Y direction,3-response of Z direction.

4.3 Dynamic decoupling and compensating experiments of the six-axis wrist force sensor

When a force is imported to the six-axis wrist force sensor, there are six channels output signals, where three channels express force components of X, Y and Z directions, and other moment components of X, Y and Z directions. To make thing simple and clear, we only consider Y and Z force channels in the paper. The decoupling between Y and Z force channels has been plotted in Fig. 5.

According to Remark 4, we can accomplish DDC of the Y and Z force channels of the sensor. Firstly, chose d=0.07824 ms, N=(4, 4), M=(2, 2). Then, applying a low-pass filter for filtering high-frequent noise and normalizing those consequent data achieve 2048 input/output pairs for training neural networks. Due to fast convergence, the *Levenberg-Marquardt* (LM) algorithm is chosen as training algorithm. Some techniques such as early stopping technique are introduced to improve the generalization of neural networks. By experiments, the nodes of the hidden layer are finally fixed as 28 and the *mean sum of squares of the networks errors* (MSE) is 0.04265.

According to Fig.3, a new sensor can be constructed when the *LISNNDDCor* is incorporated with the original sensor. Actually, the efficiency of the *LISNNDDCor* will be validated by the dynamic performance of the new sensor. Therefore, impulse force will be imported to the new sensor and its consequent response can validate if the dynamic performance is improved. Comparison experiment results between the *LISNNDDC* method proposed and VPDCN in (Xu and Li, 2000) are shown in Figs. 6 and 7 and Tab. 1.

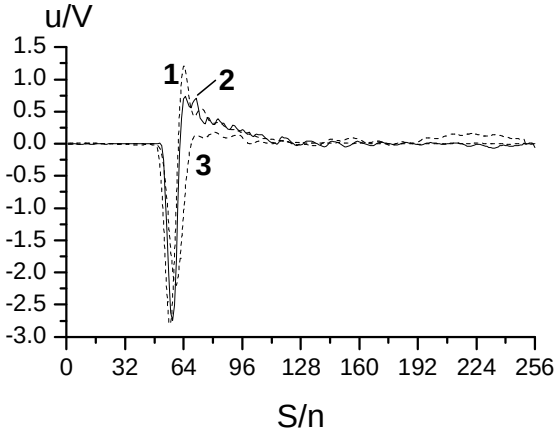

Fig. 6. DDC results of Z direction for: 1-impluse force imported from Z direction; 2-consequent response using *LISNNDDC* method proposed; and 3-consequent response using VPDCN method.

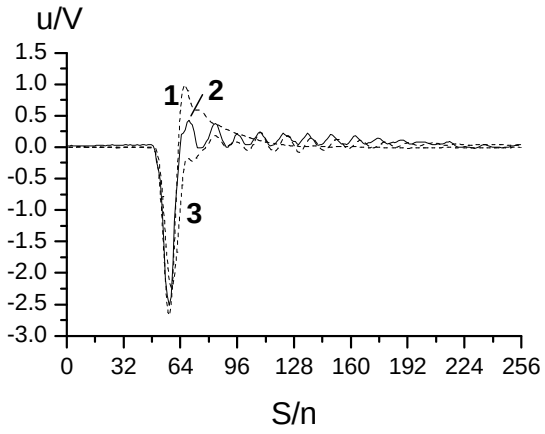

Fig. 7. DDC results of Y direction for: 1-impluse force imported from Y direction; 2-consequent response using *LISNNDDC* method proposed; and 3-consequent response using VPDCN method.

Tab.1. DDC results for Y and Z directions.

	Z direction	Y direction
LISNNDDC method δ	0.1862	0.1587
VPDCN method δ	0.4009	0.2972
LISNNDDC method E_r	1.64%	5.31%
VPDCN method E_r	21.45%	16.3%

From Figs. 6 and 7 and Tab. 1, it is easily found that using the *LISNNDDC* method proposed can obtain better dynamic decoupling and compensating performance than VPDCN method in (Xu and Li, 2000); and the dynamic performance of the chosen six-axis wrist force sensor is markedly improved.

5. CONCLUSIONS AND FURTHER WORK

A novel and practical *LISNNDDC* method is proposed for generic nonlinear muti-dimension sensors instead of linear ones. Therefore, the proposed *LISNNDDC* method is not only of prime theoretical interest but also, in practical implementation, can obtain better dynamic performance. Obviously, the *LISNNDDC* method proposed can be also used for dynamic compensation of SISO sensor system if we choose p=1.

The proposed *LISNNDDC* method has the following characteristics:

1) The principle of left-inverse theory is embedded.
2) Approximate differential algorithm is incorporated.
3) Neural networks is used for function approximation.
4) It is smart, flexible and simple. In practice, we only need to choose d, $\mathbf{N}=(n_1,n_2,\cdots,n_p)$, $\mathbf{M}=(m_1,m_2,\cdots,m_p)$, and other parameters on neural networks such as the nodes of the hidden layer, training algorithm and the techniques to improve generalization and training speed etc..

The further work can be described as follows. If p, $\mathbf{N}=(n_1,n_2,\cdots,n_p)$ and $\mathbf{M}=(m_1,m_2,\cdots,m_p)$ are very large, the number of data points in the training set is very large. For examples, for six-axis force sensor, if we choose p=6, $\mathbf{N}=(2,\cdots,2)$, $\mathbf{M}=(0,\cdots,0)$, then the *18-NODES$_{hidden\text{-}layer}$-6* network will be designed; if p=6, N=(4,$\cdots$,4), M=(2,$\cdots$,2), then the *42-NODES$_{hidden\text{-}layer}$-6* network will be designed. In addition, if d is very small, the number of data points in the training set will be very large. In such circumstances, how to improve training speed and generalization of neural networks is a key problem. Another work is to seek for practical, robust and exact high-order differential algorithm, which can substitute for the current finite difference algorithm in the paper.

REFERENCES

Liu S.A. and Tzo H.L. (2002). A novel Six-component force sensor of good measurement isotropy and sensitivities *Sens. Actu. A* **vol. 100**, pp.223-230.

Perry D.M. (1997). Multi-axis force and torque sensing *Sens. Rev.* **vol. 17, no. 2,** pp. 117-120.

Chao L.-P. and Chen K.-T. (1997) Shape optimal design and force sensitivity evaluation of six-axis force sensors. *Sens. Actuators A, Phys.* **vol. 63**, pp. 105-112.

Dubious M. (1981). Six-component stain-gge balances for large wind tunnels. *Exp. Mech.* **vol. 11** pp. 401-407.

Xu K. and Li C. (2000). Dynamic decoupling and compensating methods of multi-axis force sensors. *IEEE Trans. I & M*, **vol. 49**, pp. 935-941.

Jordan C. (1960). *Calculus of finite differences.* Chelsea Pub. Co., New York.