WPRM-CPR(18) 20378

M Li et al.

Deformation Modeling of Manipulators for DEMO Using Artificial Neural Networks

Preprint of Paper to be submitted for publication in Proceeding of 30th Symposium on Fusion Technology (SOFT)

This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 under grant agreement No 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

Deformation Modeling of Manipulators for DEMO Using Artificial Neural Networks

[1]Ming Li, [1]Huapeng Wu, [1]Heikki Handroos, [2]Robert Skilton, [2]Antony Loving

[1]Laboratory of Intelligent Machines, School of Energy Systems, Lappeenranta University of Technology, Finland
[2]Remote Applications in Challenging Environments (RACE), UKAEA, Culham Science Centre, United Kingdom

A hybrid deformation modeling method is proposed to model the deformation physics of manipulators used in DEMO. The deformation of individual joint assembly is modeled by neural network and the hybrid deformation model is constructed by integrating joint assembly's deformation with the concurrent kinematic pattern of manipulator. The Markov Chain Monte Carlo method is employed to identify weight parameters of artificial neural network. A complex joint assembly of a boom used for JET maintenance is taken as an example of applying the proposed method, which is treated as a simple one degree of freedom mechanism. The finite element method is used to generate training data for hybrid model, as well as a benchmark for the verification of the trained model. The comparison results presented in the paper indicate that the hybrid modeling method is competent to model the deformation physics of mechanisms assembly, which is kinematic dependent, under the sole force payload.

Keywords: manipulator, deep neural network, modeling, Bayesian inference, DEMO

1. Introduction

For the remote maintenance of DEMO, it is envisaged that large-scale heavy-duty manipulators will be widely employed. The payload manipulators are extreme, for instance, the multi blanket module segment (MMS) weights up to 80 tons, the divertor also weights up to 10 tons, whereas the positioning tolerance of maneuvering process of these heavy components is considerably tight, limited to 20 millimeters, compared with the large scale of 10 meters height of manipulators and the handled components [1]. From the experience of using heavy duty manipulators and robots, as well as the preliminary finite element method (FEM) simulation of conceptual DEMO manipulators, it suggests that heavy payload deforms the manipulators, and the deformation is significant in serial elongated kinematics [11]. The large deformation on one hand deviates the actual trajectories of manipulators far away from the desired positions, which may incur, in a limited space, collisions of heavy components with the surroundings, which would be catastrophic to both the surroundings and manipulators. On the other hand, the significant deformation also exert extra resistance force to the joints due to the deformed geometric profile and in worst case may degrade completely the DOF (degree of freedom) of joints, which damages the joint's drive system due to significant overload.

In order to handle the consequences of deformation, it is necessary to develop computation efficient deformation modeling methods for heavy duty manipulators. The model can predict deformation accurately in a less sensory environment, and can be incorporated into real time control system for deformation compensation. From literatures, the stiffness model of a manipulator is often developed for the deformation computation. Two main methods, namely the matrix structural analysis (MSA) and finite element analysis (FEA), are widely employed to derive a stiffness model of the objective manipulator, with their pros and cons [2-4]. In MSA, the structure of a mechanism is simplified as elementary beams, which are connected by nodes. The computation efficiency of MSA is high, however, with the compromise of accuracy, thus it is often used to evaluate qualitatively the stiffness performance of a structure or in an iterative optimization algorithm [5-7]. The FEA based method can yield accurate stress, strain and deformation results, given exact payload. However its high computation cost makes it prohibitive to apply directly for the manipulators in DEMO scenario, since the control system, as well as the iterative algorithms, require real time performance. Additionally, both methods depend on the idealized CAD model of target, and assume same material is used for all the components. By contrast in practice the manipulator assembly consists of different components, e.g. bearings, bolts, weldments, transmissions etc., as well as contains backlashes, therefore it is virtually impractical to develop the accurate deformation model from merely CAD data.

The paper proposes a hybrid deformation model for manipulators, which can utilize on-site position and payload measurements from end-effector. The joints deformation models are identified by artificial neural networks (ANNs) based on supervised training, and the hybrid model is constructed by incorporating the ANNs and concurrent kinematics of manipulator. As a matter of fact, the ANN herein represents deformation model of a virtual lump joint, which is an equivalent model of actual synthetic deformation of the joint, its fore link and backlash. The ANNs are trained by using measurements obtained on the end-effector, thus it incorporates all facts that affect final deformation outputs. The trained hybrid deformation model only consists of multiple feedforward networks and forward kinematics of manipulator, its computation can be implement in real time.

For demonstration of the proposed method, the deformation modeling of a complex joint assembly is taken as the study object in this paper, which is part of a boom used for the inside maintenance of JET, and can be deemed as a simplified 1 DOF manipulator. For training of the hybrid model, the Markov chain Monte Carlo

author's email: Ming.Li@lut.fi

(MCMC) method is employed, which can identify high dimensional parameters of neural network. To present the conducted research, the paper is organized as follows: section 2 introduces concept of a hybrid deformation modeling for generic manipulators, mathematics and training procedure; section 3 presents application of hybrid modeling for a complex joint assembly; section 4 presents the modeling results and discussions; and section 5 brings forward conclusions and future research.

2. Hybrid Deformation Modeling of Manipulators with ANNs

Artificial neural networks (ANNs) have a vast applications in modeling of linear and non-linear systems [8-10]. It has been applied to model deflection of hydraulic boom structure as a black box in [11]. The deformation modeling performance of both deterministic and Bayesian ANNs for single structural-complex joint has also been analytically investigated in [12]. The ANNs has also been widely applied to model the hot deformation of metallic alloy in material physics [13-14]. For a supervised network modeling a target system, the conventional way is to train the network under a set of input and output data, which are the actual input-output measurements of a target system. The challenging part of applying this method is the selection of the proper size and type of networks for a complex system, when the sufficient knowledge of the system is not available — if the network is constructed too small, it may be insufficient to represent the physical insight of the system; on the contrary, if the network over fits the problem, it may approximate all the noises and errors of measurements, which disguises the underlying physics. Both imperfect networks result in a poor generalization performance in the untrained domain.

Nevertheless, in practice we often have obtained comprehensive knowledge of major parts of a complex system, with only some parts of unknowns. In such scenario, a rational way to model a complex system is to use networks to identify merely unknown parts, since it is much easier to tune a network for a smaller system. A hybrid model of a complex system can be derived by integrating the network models with obtained knowledge.

In the deformation process of a heavy duty manipulator with elongated kinematics (which is likely the kinematic pattern of DEMO manipulators), the final end-effector displacement under payload is the result of a synthetic effect of deformation of individual joints and links, backlashes of assembly, and manipulator kinematics pattern. It is intractable to speculate a proper large size of neural network to model the deformation of whole manipulator as a black box, since it contains too many components and uncertainties. However, it is feasible to develop a proper size neural network for the deformation of a single joint assembly, whereas the network approximates an equivalent virtual and lump deformation model of the joint, its fore-link, and the transmission backlashes. Each joint can be modeled by such a single neural network, and these concurrent networks can then be integrated with the kinematics of manipulator to form a hybrid deformation model. Fig.1 shows the scheme of hybrid deformation modeling method.

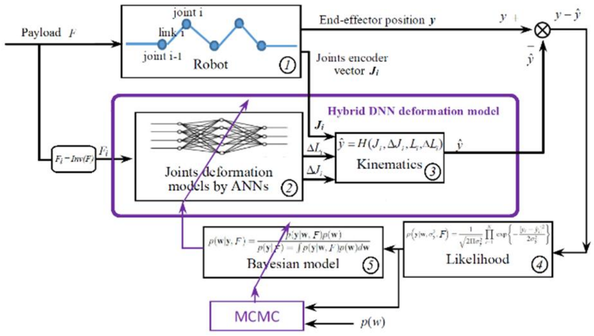


Fig.1 Hybrid deformation modeling of manipulator

Block ① represents realistic manipulators or robots, with external payload F exerted on its end-effector, whilst the resultant end-effector positon y is measured. Small size ANN in block ② is adopted herein for modeling lumped deformation of individual joint, addressing the synthetic effect of multi facts (highly nonlinear geometric profiles and backlashes) on the deformation physics. Herein the inputs of ANN consist of the force Fi vector acting on the joint, which can be obtained through the inverse computation of end-effector payload F.

The measurable data for training of hybrid model only consists of end-effector pose vector y and payload F, thus it avoids the need of measuring directly the deformation pose of joints and links, which is impractical in hostile environments. The hybrid kinematics model of manipulator which incorporates deformation models of K components (joints) is represented in block ③ by Eq. (1),

$$\hat{y} = H(J_i, \Delta J_i, L_i, \Delta L_i) = \begin{bmatrix} \sum_{i=1}^{K} (L_i + \Delta L_i + \Delta J_{i-1} \times L_i) \\ \sum_{i=1}^{K} J_i + \Delta J_i \end{bmatrix}, \quad (1)$$

where

$$\begin{bmatrix} \Delta L_i \\ \Delta J_i \end{bmatrix} = ANN(F_i) \quad (2)$$

represents the equivalent translational and orientation deformation vector of each joint i, which is the output of ANN. In Eq. (1) $\hat{y}$ represents the predicted end-effect position and orientation of hybrid model; L_i represents the geometric vector of joint's fore-link i, which is a function of joint's orientation vector J_i; ΔJ_{i-1} represents orientation deformation vector of the previous joint i-1;

For the training of hybrid model, the Bayesian inference approach is developed as in block ④ and ⑤, whereas the errors between model prediction $\hat{y}$ and the actual observation y, in Fig.1, are taken as the model residual ε taking form of independent and Gaussian noise, and the hybrid model with residual error can be represented as in

$$y = \hat{y} + \varepsilon = H(F, w) + \varepsilon, \quad (3)$$

where $H(F, w)$ is hybrid kinematics function described by external loads and joints ANNs; w represent the weights vectors that characterize ANNs in hybrid model.

The aim of Bayesian training is to infer the weights posterior probabilities, given the observation data. The corresponding Bayesian formula is represented by

$$p(w \mid y, F) = \frac{p(y \mid w, F)\, p(w)}{p(y \mid F) = \int p(y \mid w, F)\, p(w)\, dw}, \tag{4}$$

where $p(w)$ is the prior weights distribution, $p(y \mid w, F)$ is the likelihood function that gives the probability observation of *y* when given the parameters value *w* and input *F*; The most likely values of the ANNs weights *w* are those that give high values of the posterior distribution $p(w \mid y, F)$. Since the prior distribution on the observation $p(y \mid F)$ which doesn't depend on *w*, is a constant, Eq. (4) can be further represented in the form of proportionality

$$p(w \mid y, F) \propto p(y \mid w, F)\, p(w) \cdot \tag{5}$$

The Gaussian likelihood function which incorporates the hybrid model information is represented by

$$p(y \mid w, F) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \prod_{i=1}^{N} \exp\left\{ -\frac{[y - H(F, w)]^2}{2\sigma_y^2} \right\} \tag{6}$$

with zero mean, variance σ_y^2 and dimensions *N*.

To generate the high probability sequence of random weight variables *w*, whose empirical distribution can asymptotically approach to the posterior distribution $p(w \mid y, F)$, the adaptive Metropolis (AM) algorithm is used in MCMC [15]. The advantage of AM herein is that the proposed prior weights distribution is updated according to the estimated posterior covariance matrix of ANN weights, whereas the posterior covariance matrix of each simulation is computed from the past simulations, thereafter, the proposed prior distribution is updated on the knowledge learnt so far from posterior distribution. The proposal weights covariance matrix C_i can be computed, as in

$$C_i = \begin{cases} C_0 & , i \le i_0 \\ s_d Cov(w_0, \dots, w_{i-1}) + s_d e I_d & , i > i_0 \end{cases}, \tag{7}$$

where the covariance C_i has a fixed value C_0 for the first i_0 simulations; s_d is a scaling parameter, and computed by $s_d = (2.4)^2 / d$, where d is the dimension of parameter w; *e* is a small parameter used to ensure the non-singularity of C_i; I_d is the d-dimensional identity matrix.

For $i > i_0$, the computation of covariance at simulation *i+1* satisfies the recursive formula as in

$$C_{i+1} = \frac{i-1}{i} C_i + \frac{s_d}{i} \left(i \overline{w}_{i-1} \overline{w}_{i-1}^T - (i+1) \overline{w}_i \overline{w}_i^T + \overline{w}_i \overline{w}_i^T + e I_d \right), \tag{8}$$

where $\overline{w}_i = 1/(i+1) \sum_{x=0}^{i} w_x \cdot$

The detailed procedures of implementing the AM algorithm in MCMC are described as follows:

1). Initialize the algorithm with an arbitrary set of weights w_0, and an arbitrary C_0 is obtained in the form of

$$C_0 = s_d \begin{bmatrix} \sigma_{w_1}^2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_{w_2}^2 \end{bmatrix}, \tag{9}$$

where $\sigma_{w_1}, \dots, \sigma_{w_d}$ are standard deviations of the weights, which are initially set to certain arbitrary positive values;

2). Compute C_i for current simulation;

3). Generate a candidate vector w^* for *w*, from candidate distribution $p(w^* \mid w_{i-1}) = N(w_{i-1}, C_i)$;

4). Calculate the acceptance probability, α, of proposed candidate:

$$\alpha = \min\left\{ 1, \frac{p(y \mid w^*)\, p(w^*)}{p(y \mid w_{i-1})\, p(w_{i-1})} \right\}; \tag{10}$$

5). Generate $u \sim U[0,1]$, if $u < \alpha$, accept $w_i = w^*$, otherwise, set $w_i = w_{i-1}$;

6). Repeat from step 2 for next simulation until the designated simulation iterations *K* has been reached.

The identified value of w is calculated as a mean by

$$\overline{w} = 1/(K-k) \sum_{i=k+1}^{K} w_i, \tag{11}$$

where *k* represents the initial simulations that need to be discarded to diminish the effects of initial distribution.

3. Application of Deformation Modeling for a Joint Assembly

For demonstration of the proposed method, the deformation model of a complex joint assembly of a heavy duty manipulator has been developed and presented in this section. Fig.2 shows the manipulator prototype, named TARM, which has been developed in RACE, UKAEA, for remote maintenance of JET.

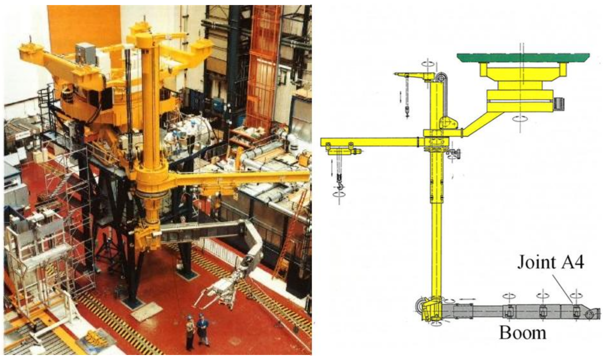
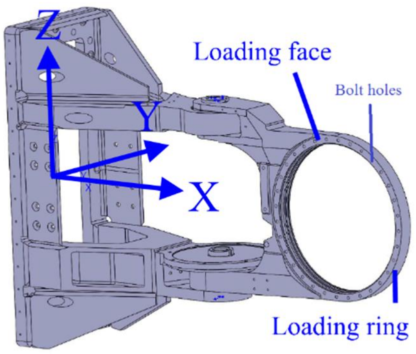
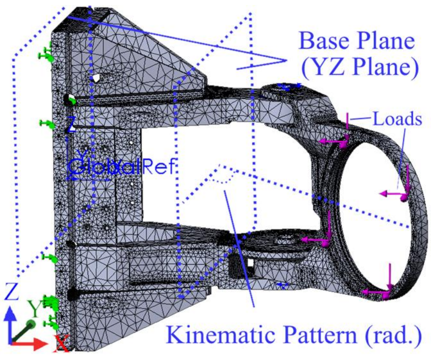


Fig. 2. TARM in-situ (left) and its schematic representation (right)

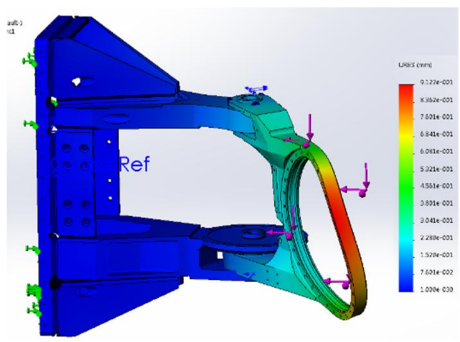
The second last joint A4 and its fore and aft links of the boom equipped on the TARM are taken as the study object due to its typical representation. Fig.3 shows its CAD assembly as well as the analysis under the FEM.



a) CAD model



b) Mesh



c) Deformed configuration

Fig.3 CAD model of A4 joint assembly and its analysis

TABLE I
DEFORMATION DATA OBTAINED USING FEM

[a]Kinematics (unit: rad)		[b]Loads (unit: kN)			[c]Deformation (unit: micrometre)	
1.3090(No.31)	-0.4590	0.3820	-0.9090	8.7898	31.6618	-57.6449
1.3265(No.32)	-0.9780	0.8740	-0.9020	19.7714	82.9644	-55.9991
1.3439(No.33)	-0.0240	0.5450	-0.9620	20.0673	74.8784	-59.2806
1.3614(No.34)	0.8950	-0.7430	-1.9610	-20.4327	-83.0682	-125.5333
1.3788(No.35)	-0.0320	-0.9000	-0.7150	-38.3505	-141.0755	-48.9571
1.3963(No.36)	0.4250	-0.8010	-1.0090	-23.2071	-109.8341	-66.0743
1.4137(No.37)	-0.2800	-0.6860	-2.4150	-28.3082	-125.2850	-153.3982
1.4312(No.38)	0.8790	0.9000	-2.7950	31.5201	155.3869	-170.0306
1.4486(No.39)	-0.9720	-0.0800	-1.9770	-10.9289	-49.7806	-126.1084
1.4661(No.40)	-0.0720	-0.9050	-1.2430	-26.4049	-147.4567	-79.5736
1.4835(No.41)	-0.4350	0.1850	-2.1580	-0.7249	9.2359	-133.4439
1.5010(No.42)	0.4330	0.1120	-0.5920	4.0367	22.4432	-36.0642
1.5184(No.43)	-0.3420	-0.6280	-2.7020	-14.6491	-116.5216	-171.8733
1.5359(No.44)	0.0640	0.1550	-2.9320	0.8434	13.2269	-182.1910
1.5533(No.45)	0.0430	0.2190	-2.6920	0.4063	23.7218	-166.3915
1.5708(No.46)	0.6200	-0.0420	-2.3450	0.5121	-12.1161	-144.9308
1.5882(No.47)	-0.5470	-0.5420	-2.3810	-7.3022	-100.9476	-151.0883
1.6057(No.48)	0.8340	-0.6840	-2.2480	-1.2594	-115.6858	-141.5370
1.6232(No.49)	-0.8460	0.3560	-0.8520	-0.7426	51.4096	-51.4821
1.6406(No.50)	-0.0290	0.9670	-1.7240	0.5587	148.5470	-103.8747

a. Kinematic parameter represents angle between joint's fore and aft links, and No.x represents the sequence number in 91 random load and kinematic pairs;
b. Loads consist of force along x, y, and z axis in the reference frame, and the units are kN;
c. Deformation consists of deformation along x, y, and z axis in the reference frame, and the units are micrometre.

A global coordinate reference frame XYZ is set in the centre of base plane of assembly. All the measurements thereafter are expressed in this frame. The external load is exerted on the loading face of loading ring (Fig.3a), whereas the next link and joint's driving unit are attached actually through evenly distributed bolts. The kinematic pattern of this assembly is expressed by the angles (in radian) between the YZ plane of global frame and the loading face plane, as in Fig.3b.

Currently in this paper, the deformation modeling has merely been conducted in simulation environment due to realistic limitations (the TARM prototype is still under upgrading), whereas the payload effect on deformation is investigated. Nevertheless, the simulation results can still be used to verify the feasibility of proposed modeling method, whereas the FEM data are taken as benchmark.

To obtain training data for deformation modeling, 91 set of random load vectors are exerted on the face of loading ring of assembly (Fig.3), whilst the assembly is in 91 different kinematic patterns, and the ranges of x, y, z component of each load vector are [-1000, 1000] N, [-1000, 1000] N and [-500, -3000] N, respectively. The FEM analysis is conducted in Solidworks COSMOS package by a program written in VisualBasic which interfaces with COSMOS API and automates assembly's kinematic pattern reconfiguration, random force loading, meshing, analysis and measuring process. In the FEM analysis, the 1060 aluminum alloy is adopted, considering the materials adopted in the realistic structure. Global contact bonded is adopted for all the contacting components in assembly, and the joints are connected according to realistic situation, whilst the base plane of assembly is rigidly fixed. Curvature-based solid mesh is conducted with following parameters: the minimum element size equals 7.6 millimeter (mm); the maximum element size equals 38 mm; minimum element number in circle is 8; and growth ratio is 1.6. After the program has run, the mean displacement of all the nodes in loading face is used as the model deformation. Part of deformation measurements obtained under different loads are presented in Table 1.

In order to design the proper size of neural network, the deformation model of this joint assembly needs to be analyzed. The deformation kinematics of joint assembly is represented schematically in Fig.4.

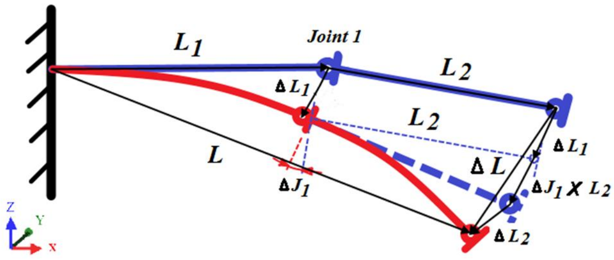


Fig.4 Schematic representation of joint assembly

The realistic complex joint connections are modeled by a virtual joint *1*. L_1 and L_2 represent the geometric vectors of fore and aft links; ΔL_1 and ΔJ_1 the translational and orientation deformation of joint *1*; ΔL_2 the translational deformation of the end-effector. Applying superposition principle, the analytical deformation model of joint assembly is expressed by Eq. (12), (13) and (14), according to hybrid deformation modeling methodology described in Section 2:

$$L = L_1 + \Delta L_1 + L_2 + \Delta L_2 + \Delta J_1 \times L_2, \tag{12}$$

where

$$\begin{bmatrix} \Delta L_1 \\ \Delta J_1 \end{bmatrix} = ANN_1(F_1), \tag{13}$$

and

$$\Delta L_2 = ANN_2(F_2). \tag{14}$$

Since the acting force F_1 in joint *1* equals approximately to F_2 acting on the end-effector of assembly, a hidden layer neural network can be used to represent the synthetic effect of Eq. (13) and (14). An extra hidden layer are needed to represent the mathematic model of Eq. (12), considering ΔL_1, ΔJ_1 and ΔL_2 as its input variables. Consequently, a small size of deep neural network (DNN) with two hidden layers is used herein to model the deformation physics of the whole joint assembly. A full connected feedforward network structure is adopted, and its scheme is presented in Fig.5.

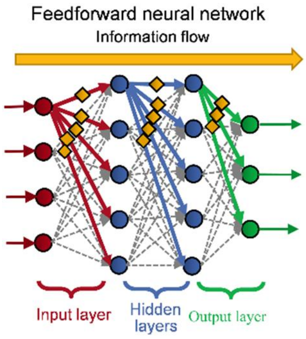


Fig.5 Deep neural network for joint assembly deformation

The DNN contains four inputs, which are the three dimensional load vector acting on the loading face of assembly and the orientation (kinematic pattern) of the joint, whist the three outputs consist of mean deformation displacement vector of all nodes in the face of loading ring. 30 neurons are adopted in each hidden layer, which is chosen through the accuracy comparison in several training tests in different neuron numbers. The sigmoid activation function is used for the neurons in hidden layers, and the linear activation function is used in the output layer. It should be noted that we can't conclude the 30 neurons herein in hidden layers are the optimal choice for the application in the paper, since how to choose and optimize the neural network size is another research topic and we didn't exhaust all possible options.

4. Results and Discussions

The constructed DNN deformation model is trained by the method (procedures) developed in Section 2. In training process, the deformation model is first trained by Levenberg-Marquardt (LM) algorithm through back propagation. The training criterions are set as: a) the root mean square errors (RMSs) reaching 0.001 or b) the training iterations reaching 1000. In the training, we observed the RMSs become stable after 100 iterations, and the training stopped at the maximum iteration setting. The LM training can provide an initial guess of DNN weights and the weights covariance matrix, which correspond to C_0 in Eq. (9) and is important to guarantee fast convergence of MCMC. Although it is likely the initial weights fall into local optimum, the MCMC based Bayesian inference can explore high probability weights in global region.

Fig.6 shows the prediction results of deformation model trained by MCMC in the trained domain, which stopped after it has generated 10000 samples in Markov chain.

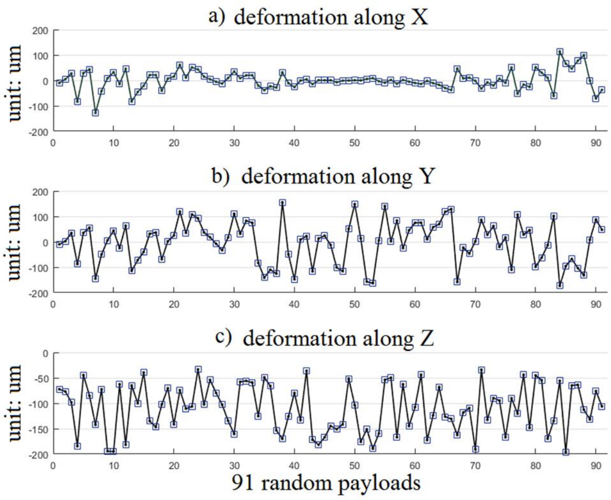


Fig.6 DNN prediction of assembly deformation in trained area

The squares in figure plots stand for the deformation data (as a benchmark) obtained by FEM, with the unit in micrometer, under 91 set of random loads, whereas the continuous lines stand for the deformation model predictions under the same loads and kinematic patterns.

After convergence of training, the root mean squares (RMSs) of errors between predictions and benchmark data are, 0.0555, 0.0970, 0.1037, respectively, along x, y and z direction of reference frame. In the Fig.6a, the joint deformation along x axis under random payload from No.41 to 65 is relatively smaller, presented as a relatively flat curve, which indicates a higher stiffness of joint assembly in x axis. The reason is that the joint's fore and aft links are in a nearly aligned kinematic patterns from No.41 to 65, which provides a stiffer pattern than others along y and z axis. Fig.6b and Fig.6c indicate that the joint assembly deformation is more affected by the loads along y and z axis, with less effect of kinematic patterns.

To verify the generalization ability of trained deformation model of joint assembly in untrained region, another 91 set of random loads are exerted to the assembly in 91 different kinematic patterns, and the corresponding comparison between measurements of FEM analysis and model prediction is presented in Fig.7.

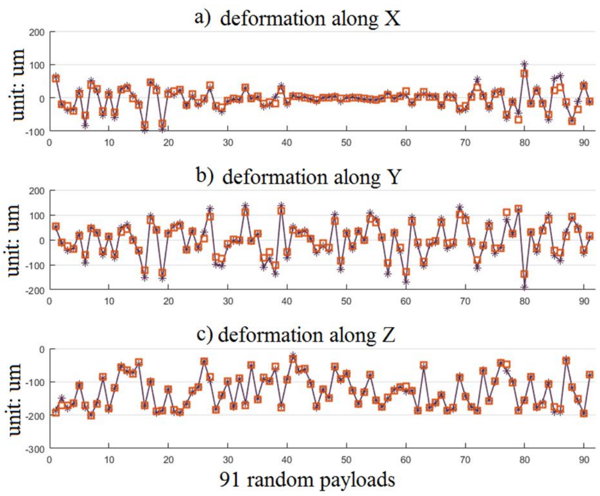


Fig.7 Prediction of assembly deformation in untrained domain

The squares in figure plots represent the deformation data along x, y and z directions obtained by FEM, whereas the continuous lines with asteroids represent the deformation predictions, respectively. The RMSs of errors between the predictions and FEM benchmark along x, y and z directions are 10.9744, 8.6666, and 5.4346, respectively. The RMSs herein are higher than those in the trained domain, nevertheless, it is still inside the rational acceptable range, which indicates that the deformation model is well developed and reveals underlying deformation physics of this joint assembly. The error difference of the DNN model in the trained domain and untrained domain is interpreted as a normal phenomenon herein. The big difference are caused by serval reasons: 1) the RMSs in the trained domain are already very small, thus the big difference from the trained RMSs doesn't suggest the prediction of model in untrained domain is erroneous; 2) the RMSs tend to give a relatively high weight to large errors, whereas there are few relatively larger errors for untrained prediction, e.g. the errors from payload number 85 and 86 in Fig.7a; 2) for a data set from an experiment, the neural networks are commonly trained with 70% of its data, and the rest 30% are used for the model verification. It should be noted that the 30% and 70% data are all from the same experiment domain. However, in this paper, in order to verify if the hybrid model can identify the underlying deformation physics, same amount of verification data to the trained data are obtained from another new experiment, and are all used for the testing (prediction performance evaluation), thus the RMSs are larger than those in the trained domain. However, through the prediction curves we could conclude the hybrid model performances well in prediction. For comparison convenience, the assembly's deformation RMSs between predictions and FEM data, in trained region and untrained region, are listed in Table 2.

TABLE 2
ROOT MEAN SQUARES OF DEFORMATION MODEL UNDER TRAINED DATA AND NEW DATA

Under Trained Data	deformation along X axis	deformation along Y axis	deformation along Z axis
RMSs	0.0555	0.0970	0.1037
Under Untrained New Data	deformation along X axis	deformation along Y axis	deformation along Z axis
RMSs	10.7944	8.6666	5.4346

5. Conclusions and Future Research

Through the developed deformation model and the corresponding comparison results of a typical 1 DOF joint's assembly of a manipulator prototype, we conclude that the hybrid modeling method is competent in modeling end-effector deformation of kinematic dependent mechanism by incorporating the kinematics information, which is under the force payload effect in this paper. The ANN, through limited training, can reveal the deformation physics of kinematic dependent mechanism under random force payload, and can extrapolate the developed prediction model to untrained domain. The degenerated prediction performance of deformation model in untrained domain also suggests that the network can be further optimized in terms of hidden layer numbers, neuron numbers, as well as adoption of more characterized training data.

It should be noted that the application of the hybrid modeling is conducted on a one DOF mechanism in the paper, whereas the number of hidden layers can be analytically deduced to represent the whole deformation physics. However, when the manipulators consist of several DOF, it is suggested several concurrent ANNs should be used to represent the joints deformation physics individually, and should be integrated into the hybrid deformation kinematics (Eq. (1) or (12)) to form deformation model of the whole manipulator, rather than using a very complex DNN to represent the whole manipulator deformation.

The deformation modeling of manipulators and robots used in DEMO and future fusion plants is by no means solved in current research stage. The environmental loads that include varying thermal effect, high neutron radiation around fusion reactor, will change the material deformation physics in a coupled way with force loads. The extreme force payload also introduces more complex transmission systems applied in the mechanical design, which bring more uncertainties in terms of backlashes, transmission deformation etc. The convolutional neural network will be investigated in the future research to filter and decouple the synthetic effects of various environment loads, and the recurrent neural network will also be studied to model the time sequential effect of those loads. Regarding to the difficulties of obtaining the measurement data manually in a fusion plant environment, the unsupervised learning will be investigated to model and compensate the manipulators and robots deformation autonomously by interacting with the surrounding environments.

Acknowledgements

This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 under grant agreement No 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

References

[1] J. Keep etc., Remote handling of DEMO breeder blanket segments: Blanket transporter conceptual studies, Fusion Engineering and Design, 2017.

[2] A. Ghali, A. Neville, T. Brown, Structural analysis: a unified classical and matrix approach, CRC Press, 2003.
[3] C. Gosselin. Stiffness mapping for parallel manipulators, IEEE Transactions on Robotics and Automation, Vol. 6, No. 3, pp. 377-382.
[4] B. Thompson etc. A survey of finite element techniques for mechanism design, Mechanism and Machine Theory, Vol. 21, Issue 4, 1986, pp 351-359.
[5] M. Li, H. Wu, H. Handroos. Static stiffness modeling of a novel hybrid redundant robot machine, Fusion Engineering and Design, Vol. 86, Issue 9-11, pp. 1838-1842, 2011.
[6] M. Li, H Wu, H. Handroos. Stiffness-maximum trajectory planning of a hybrid kinematic-redundant robot machine, IECON 2011, Melbourne, Australia, 2011.
[7] M. Li, Stiffness based trajectory planning and feedforward based vibration suppression control of parallel robot machine, doctoral dissertation, 2014.
[8] S.R. Chu, R. Shoureshi, M. Tennorio, Neural networks for system identification, IEEE Control Systems Magazine, Vol. 10, Issue 3, April 1990.
[9] S. Chen, S.A. Billings, Neural networks for nonlinear dynamic system modelling and identification, International Journal of Control, Vol. 56, Issue 2, pp. 319-346, March 2007.
[10] M. Li, H. Wu, Y. Wang, H. Handroos, G. Carbone, Modified Levenberg-Marquardt algorithm for backpropagation neural network training in dynamic model identification of mechanical systems, Journal of Dynamic Systems, Measurement, and Control, Vol.139, March, 2017.
[11] A. Rouvinen, Use of neural networks in robot positioning of large flexible redundant manipulators, doctoral dissertation, 1999.
[12] M. Li, H. Wu, H. Handroos, R. Skilton, J. Keep, A. Loving, Comparison of deformation models of flexible manipulator joints for use in DEMO, IEEE Transactions on Plasma Science, Issue 99, 2018.
[13] N. Haghdadi, A. Zarei-Hanzaki, A. R. Khalesian, H. R. Abedi, Artificial neural network modeling to predict the hot deformation behavior of an A356 aluminum alloy, Materials & Design, vol. 49, pp. 386-391, August 2013.
[14] R. Kapoor, D. Pal, J.K. Chakravartty, Use of neural networks to predict the deformation behaviour of Zr-2.5Nb-0.5Cu, Journal of Materials Processing Technology, Vol. 169, Issue 2, pp. 199-205, November 2005.
[15] H. Harrio, E. Saksman, and J. Tamminen, An adaptive metropolis algorithm, Bernoulli, 7(2), pp. 223-242.