Microsystem Technologies (2020) 26:209–218
https://doi.org/10.1007/s00542-019-04554-5

TECHNICAL PAPER

Position error prediction using hybrid recurrent neural network algorithm for improvement of pose accuracy of cable driven parallel robots

Jung-Min Kang[1,2] · Sung-Hyun Choi[1,2] · Jung-wan Park[1,2] · Kyoung-Su Park[1,2]

Received: 4 December 2018 / Accepted: 18 July 2019 / Published online: 25 July 2019

Abstract

Because cable-driven parallel robots (CDPRs) have lightweight moving parts, CDPRs have been used in various industrial applications requiring high speeds and accelerations. Especially, CDPRs with polymer cables can achieve higher accelerations and speeds compared to those with steel cables. However, they also have some disadvantages, such as a nonlinear creep, a hysteresis, and a short- and long-term recovery. Because these nonlinear characteristics, the accuracy of CDPRs gets worse and worse. In this study, we proposed a hybrid recurrent neural network (H-RNN) to predict nonlinear characteristics of the cable elongation and to solve the problems associated with CDPRs and apply the real-time control. In the algorithm, the long short-term memory (LSTM) algorithm was used to learn the characteristics of the low-frequency data, and the basic RNN learned the features of the high-frequency data. We also confirmed that the cut-off frequency was determined based on the non-operating frequency related to rest time. Also, it yielded more accurate results because the LSTM has a wider effective frequency range. Finally, the learning process was completed by combining these two algorithms. These results made it possible to predict position errors of CDPRs with high accuracy, in which error varies under both while operating and no operation conditions. The H-RNN had a lower root mean square error than both the optimal RNN and the optimal LSTM, so it was effective for controlling systems that have errors across a range of frequencies.

1 Introduction

Cable-driven parallel robots (CDPRs) have lightweight moving parts, including very small end effectors and light cables rather than rigid manipulators. Hence, CDPRs have been used in various industrial applications that require high speeds and accelerations. Due to flexible cables, it is easy to expand the workspaces of CDPRs by widening only their frames, so they have been used in applications that require large workspaces. Indeed, they have been used for a variety of applications, such as skycams, surgical robots, and phage robots (Hannaford et al. 2013). These require a high mass efficiency, large workspace, and high levels of accuracy (Williams II et al. 2008; Merlet and Daney 2010; Gobbi et al. 2011) for long periods of time.

Most CDPRs use steel or polymer cables to actuate the end effector. Although CDPRs with wired steel cables can support high loads, they are not suitable for application in fast systems because their Young’s modulus can be changed significantly by applied tension, and they have high weights and bending moments (Hyun Dong 2017). In contrast, CDPRs with polymer cables can achieve high accelerations and speeds. They have therefore been used for applications that require fast responses, such as skycams. Although polymer cable robots have many advantages, such as being light in weight, having high flexibility, etc. However, they have some disadvantages, such as nonlinear creep, hysteresis, and inaccuracies due to short- and long-term recovery (Chattopadhyay 1997). Creep occurs when tension is applied to the cable for a short time, and affects up to 15% of the cable length. Hysteresis is the phenomenon that causes the cable length to vary differently during loading than during unloading. This is caused by the

✉ Kyoung-Su Park
pks6348@gachon.ac.kr

[1] Department of Mechanical Engineering, Gachon University, 1342 Seongnamdaero, Sujeong-gu, Seongnam-si, Gyeonggi-do 461-701, Republic of Korea

[2] Hyundai Motor, 150, Hyundaiyeonguso-ro, Namyang-eup, Hwaseong-si, Gyeonggido, Republic of Korea

differences between stretching and shrinking. Short-term recovery is the rapid restoration of the cable length, within a few seconds, upon completing an unloading operation.

These processes take between a few milliseconds and a few seconds. As a result, this type of recovery occurs with relatively high frequency. In contrast, long-term recovery occurs over a few days, and thus has relatively low frequency. Similarly, the accuracy of a CDPR system has high-frequency and low-frequency components due to the non-linearity of the properties of the cable.

Many studies have been carried out to overcome these uncertain, nonlinear behaviors. Pott (2017) and Merlet (2009) applied a modified version of Hooke's law to solve the dynamic creep. Miyasaka et al. (2016) developed a longitudinally stretched cable model consisting of a damping system and hysteresis that can be used to control a cable driven machine. These researchers focused on individual, simplified characteristics. In our previous research (Choi and Park 2018), we proposed an integrated nonlinear dynamic cable model to solve all of these problems easily. This was the most accurate model with respect to all of the nonlinear properties of CDPRs. However, the main problem with our integrated nonlinear dynamic model was its complexity and high computational cost, which make it difficult to apply the model to real CDPR control systems.

Recently, a breakthrough was made in the prediction of nonlinear characteristics via the development of a model that uses an artificial neural network (ANN) (Wang et al. 2014; Yan and Wang 2014). This approach can easily overcome the complexities associated with existing nonlinear models. Levin and Narendra (1996) simplified the complex nonlinearities using an ANN, and Anderson (1989) used an ANN for action and evaluation functions, thus improving the control of the devices. Jung et al. (2008) proposed solving the nonlinear inverted-pendulum problem by introducing an ANN to the signal processing stage and performed real-time control with a proportional integral derivative (PID) controller. Li et al. (2010) solved uncertainties in the CDPR using an ANN. However, as mentioned above, the nonlinear characteristics of the cable have both short-term and long-term variations, so it is difficult to apply this approach to CDPR systems directly.

In this study, we investigated the application of a hybrid recurrent neural network (H-RNN) to nonlinear modeling and control to solve the problems associated with CDPRs and real-time control applications. We used a hybrid frequency-based learning method to clarify the complex nonlinear characteristics of polymer cables. We investigated the effectiveness of the hybrid frequency-based RNN method by first constructing a CDPR system, and then using a stereo webcam to measure the position errors. We designed a 12-point trajectory that reflects the characteristics of the system in all directions. The position errors were measured at each point and the results were used as the test dataset. We then developed a hybrid RNN learning algorithm to estimate the measured error. The long short-term memory (LSTM) algorithm was used to learn the characteristics of the low-frequency data, and the basic RNN learned the features of the high-frequency data. Finally, the learning process was completed by combining these two algorithms. The final error data were predicted with the same process. We compared the result obtained from the H-RNN with the results from the optimal LSTM and RNN algorithms.

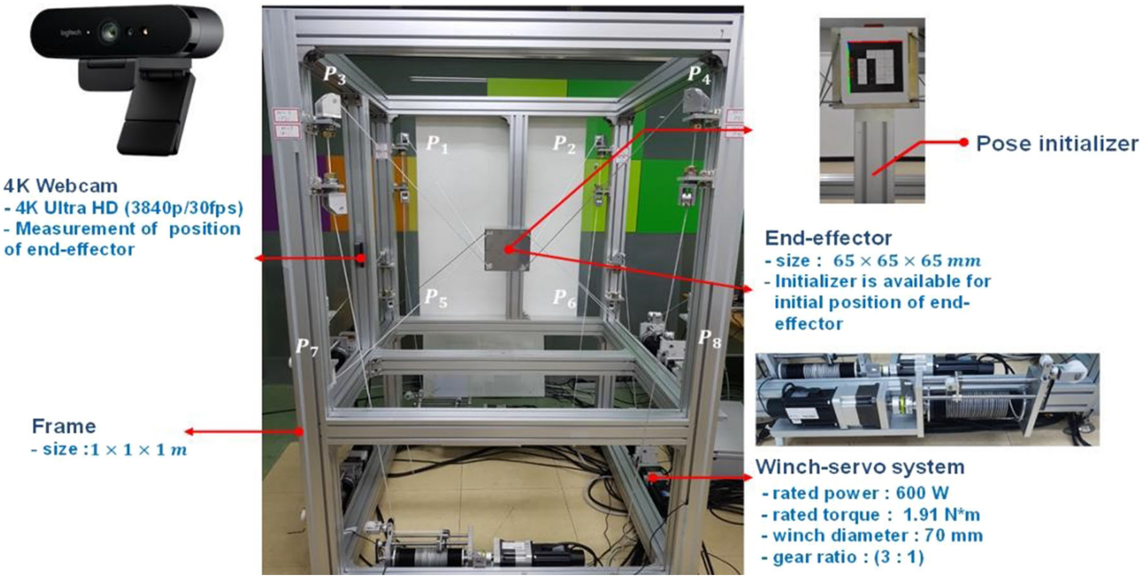

P_3
P_4
P_1
P_2
P_5
P_6
P_7
P_8
4K Webcam
- 4K Ultra HD (3840p/30fps)
- Measurement of position of end-effector
Pose initializer
End-effector
- size : $65 \times 65 \times 65$ mm
- Initializer is available for initial position of end-effector
Frame
- size :$1 \times 1 \times 1$ m
Winch-servo system
- rated power : 600 W
- rated torque : 1.91 N*m
- winch diameter : 70 mm
- gear ratio : (3 : 1)

Fig. 1 Cable-driven pulley robot system used to construct the data set (Choi 2018)

2 Construction of system setup and data acquisition

2.1 Experimental setup of the CDPR with 8 cables

As shown in Fig. 1, CDPR with a 1 × 1 × 1 m^3 workspace was constructed and training data were obtained through the constructed experimental setup. We used the Dyneema SK78 model with polyethylene cable. The construction of the CDPR was fully constrained, with 8 cables and 6 degrees of freedom. Each cable was guided by a series of 4 pulleys, and each cable was controlled by a winch-servo system. The end effector had dimensions 65 × 65 × 65 mm^3. Before operating the CDPR, we constructed a pose initializer to fix the end effector to its initial position. We measured the end effector pose estimation error as the end effector moved using a 4 K webcam (380*p*/30*fps*) and a marker detection algorithm. Our CDPR system was operated using an accurate position control algorithm consisting of inverse kinematics, a compensation function for the effect of friction on the pulley, and the pulley kinematics (Pott 2012).

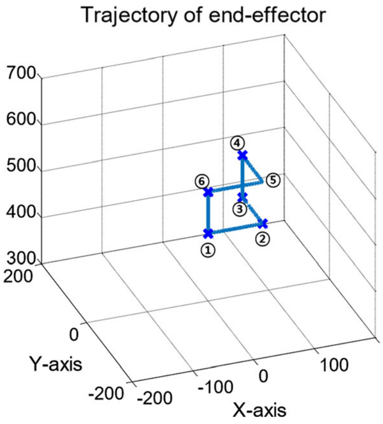


Fig. 2 The trajectory used to construct the training and test sets

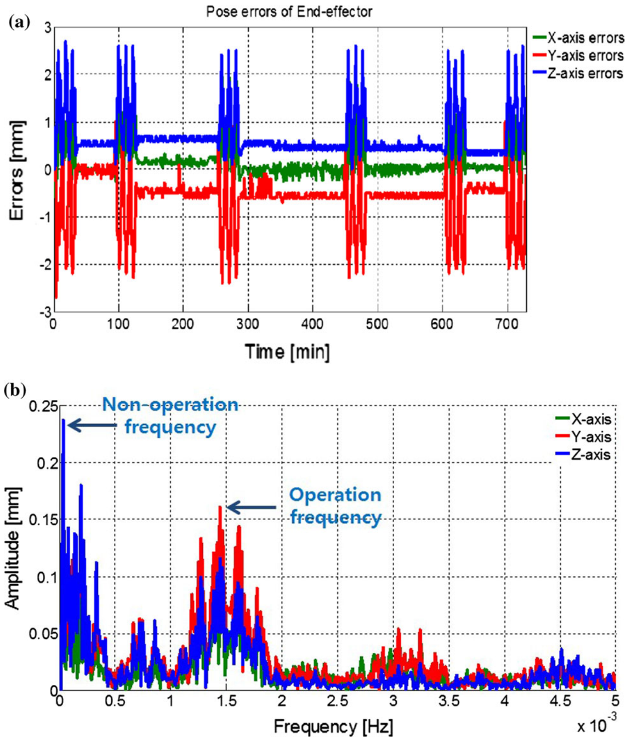
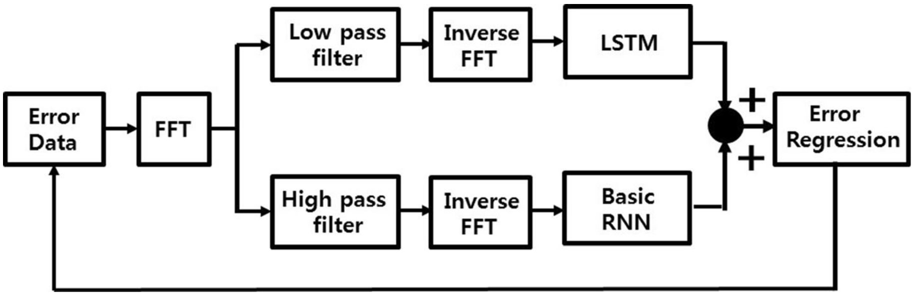


Fig. 4 Training set based on the end effector errors: **a** time domain-based training set, **b** frequency domain-based training set

2.2 Construction of the training set and test set based on the experimental results

The training and test sets, that are the position error data were constructed by operating the CDPR along a cubic trajectory of dimensions 90 × 90 × 90 mm in the *x, y* and *z* planes, as shown in Fig. 2. Because one or two certain cable can have relatively high or low tensions when moved by more than 90 mm, the trajectory was made within 90 mm. The cubic trajectory consists of 12 points in total, and the data were measured at each point every 60 s, which is long enough for sufficient creep to occur along the cable. The CDPR was operated with the same manner three or four times to observe the hysteresis and short-term recovery, which we measured every 15 s. After this operation, the long-term recovery was measured by unloading the CDPR which maintains a low tension for 1–2 h. For

Fig. 3 Framework of frequency-based hybrid recurrent neural network



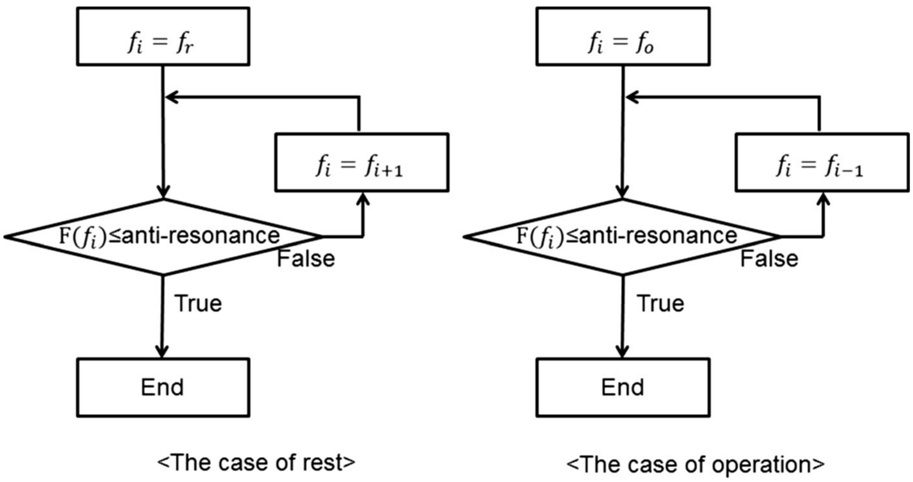


Fig. 5 The flow chart for evaluating the optimal frequency division

unloading, position errors were measured every 2 min because the conditions varied very slowly. As the end effector moved along the determined trajectory, we measured the pose estimation errors for *x, y,* and *z*-axes at each position. The root mean square error (RMSE) induced by the controller was investigated as 0.23 mm. Hence, we excluded the effect of the very small controller error from our analysis. The inputs to the training set consisted of tension sets, which included the factors that influence the cable length, loading time, and unloading time (Choi and Park 2018). Totally measured position error data was 729, which was determined with the minimum learning error based on our experiment. The learning engines used were Tensorflow and Keras models. In the model, the hyper parameters used the gradient descent method using Adam optimizer with the best performance, learning rate, number of layers and look back coefficient. Also, the learning rate was used until the cost converged to a certain value and was no longer affected by learning, and 0.001 was used in this paper. The number of layers is experimentally used, and the value is 30. The look back coefficient is now a parameter that determines how much of the previous data should be used to predict the next data, and a value of 3 was used. Among total data, 70% of the data were used as training sets and 30% of the data were used as test sets. The input used for the prediction was eight tension sets of CDPRs and the label is position errors. The error of the next state can be predicted by the tension and error of the previous stage.

3 Frequency-based H-RNN for CDPRs

This section presents a frequency-based H-RNN that can take into account all of the nonlinear phenomena that arise during operation (loading condition) and when the CDPR is not operating (unloading condition). In our previous research (Choi and Park 2018), change of the applied

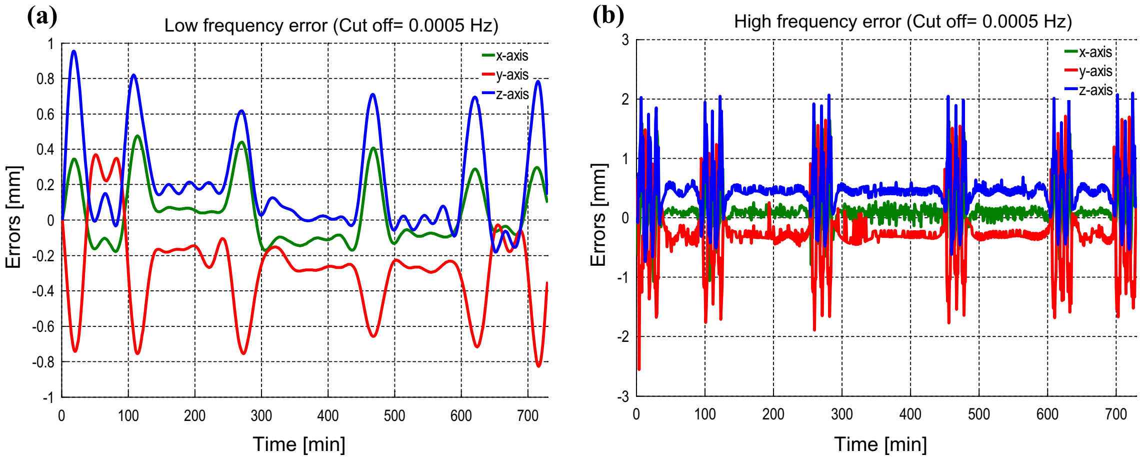


Fig. 6 The measured data, divided with a cut-off frequency of 0.0005 Hz: **a** low-frequency region (∼ 0.0005 Hz), **b** high frequency region (0.0005 ∼)

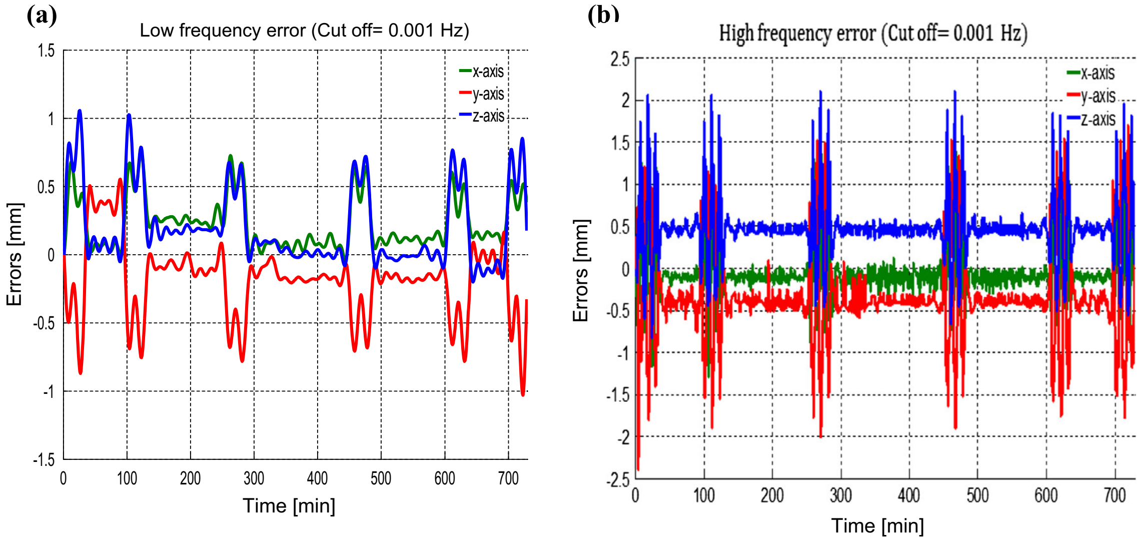


Fig. 7 The measured data, divided with a cut-off frequency of 0.001 Hz: **a** low-frequency region (0–0.001 Hz), **b** high-frequency region (0.001 Hz ~)

tension dominantly affects a long-term recovery, and causes elongation of the cable. Therefore, in the non-operating period, the change in applied load caused position errors with relatively low frequency due to the long-term recovery. On the other hands, errors occurred when the applied tension was changed by the position and dwell time and time of short-term recovery were changed during operation. These errors occur with relatively high frequency because they vary within seconds. High-frequency phenomena, such as creep, hysteresis, and short-term recovery, are more accurately represented by a basic RNN, whereas low-frequency phenomena, such as long-term recovery, are more accurately learned using LSTM because the LSTM acts as a low pass filter (Le and Zuidema 2015; Bengio et al. 2013). Figure 3 shows the framework for the H-RNN. The errors (i.e., the cable length errors and end-effector errors) generated by the CDPR system are first converted to the frequency domain by the fast Fourier transform (FFT). The modified error data are divided into operating frequency (high frequency) and non-operating frequency (low frequency) regions according to the cut-off frequency, which we determined based on the operating conditions and trajectory. The criteria used to determine the cut off are explained in Sect. 4. The features of the divided training set were individually then learned by the LSTM and RNN, and then the results are added arithmetically. The LSTM learned the features of the training set for the non-operation domain, which was dominated by low-frequency of long-term recovery. The RNN analyzed the training set representing the high-frequency (operating) components, which are generated by the fast movement of the end-effector in various directions. Finally, we obtained the errors by combining the results obtained from the two algorithms.

Table 1 Optimal parameters for the *x*, *y*, and *z*-axes for the LSTM with the softsign function and the RNN with the tanh function

	x	*y*	*z*
Under 0.0005 Hz (LSTM: softsign)			
Sequence length	11	11	11
Hidden dim	14	14	14
Over 0.0005 Hz (RNN: tanh)			
Sequence length	13	13	13
Hidden dim	14	14	14
Under 0.001 Hz (LSTM: softsign)			
Sequence length	13	13	13
Hidden dim	14	14	14
Over 0.001 Hz (RNN: tanh)			
Sequence length	13	13	13
Hidden dim	14	14	14

4 Learning and discussion

4.1 Determination of the cut-off frequency of the H-RNN

Figure 4a shows the pose estimation error of end effector over time. The green, red, and blue lines indicate the pose errors in the *x*, *y*, and *z*-axis, respectively. The CDPR was operated in a cubic trajectory for three cycles. The CDPR then rested in a non-operational state for 1–3 h after unloaded. Figure 4b shows the results of the FFT for each set of experimental results. There are two methods for determining the cut-off frequency: one method is based on the operating speed and the other is based on the rest time.

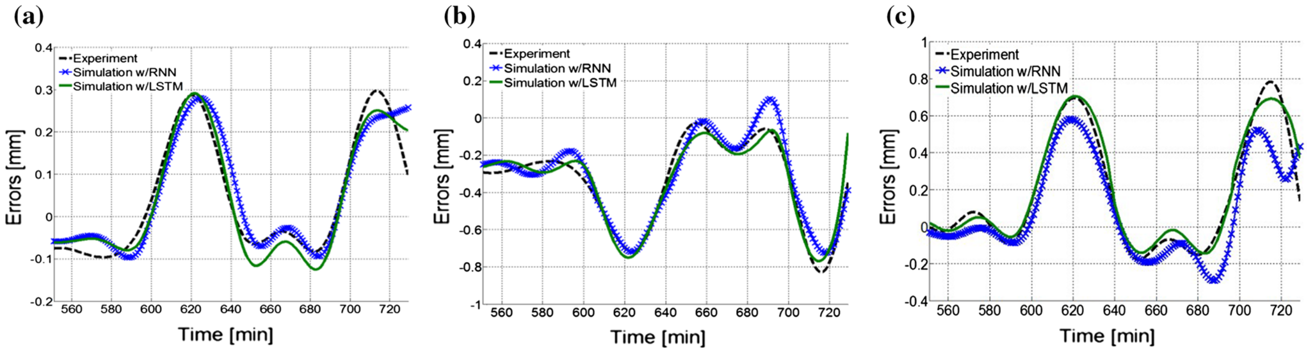


Fig. 8 Learning results for the low-frequency components based on a cut-off frequency of 0.0005 Hz: **a** *x*-axis, **b** *y*-axis, **c** *z*-axis

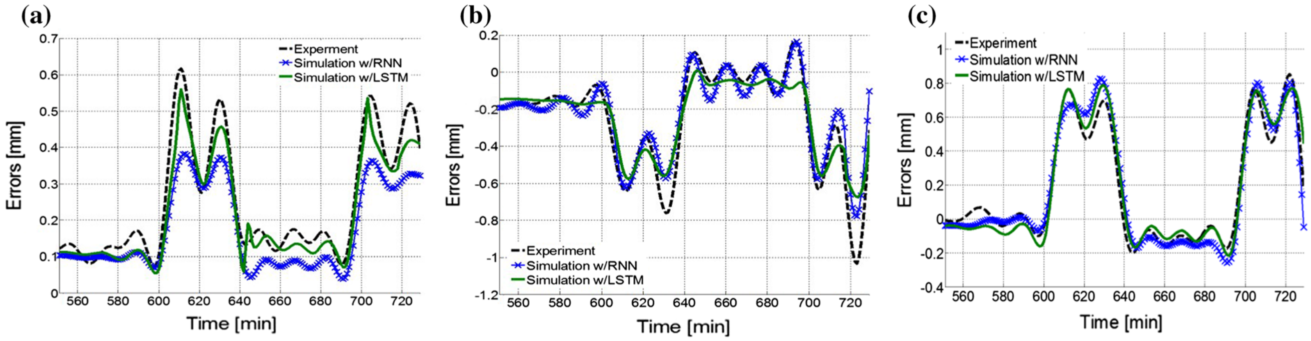


Fig. 9 Learning results for the low-frequency components based on a cut-off frequency of 0.001 Hz: **a** *x*-axis, **b** *y*-axis, **c** *z*-axis

Table 2 Root mean squared error of the low frequency components (mm)

RMSE (mm)	Axis	LSTM	RNN	RNN/LSTM (%)
0–0.0005 Hz	*X*	0.033	0.046	139
	Y	0.039	0.071	182
	Z	0.059	0.158	267
0–0.001 Hz	*X*	0.190	0.238	125
	Y	0.121	0.189	156
	Z	0.071	0.091	128

The sample datasets contained 12 points per a minute cycle, and the main operating frequency was 0.00138 Hz. The frequency of the long-term recovery and static creep, which occurred in the static (non-operating) state, was close to 0 Hz. We determined the cut-off frequency between the operating and non-operating states so that we could apply the frequency-based H-RNN algorithm.

Figure 5 presents a flow chart of the process used to divide the frequency. We applied two methods and compared the results, then determined the most effective frequency division criterion for minimizing the loss of data. One method is to divide the frequency based on the non-operating state related to long-term recovery time, and the other is based on the operating frequency. In the flow chart, f_r and f_o are the frequencies with maximum rest amplitude and operation interval, respectively. Each frequency is classified and the amplitude is calculated using the FFT. We determined the cut-off frequency as being the anti-resonance frequency of the operating and non-operating frequencies, the amplitude of which is approximately zero. When using the non-operating frequency (rest condition) for reference. the cut-off frequency increases until this condition is satisfied. When using the operating frequency for reference, the cut-off frequency is decreased until the condition is satisfied. According to these methods, the cut-off frequency was either 0.0005 or 0.001 Hz, as shown in Figs. 6 and 7, respectively. Figure 7 has much more data in the low-frequency region than Fig. 6. While Fig. 6 has a few low-frequency fluctuations, all high-frequency terms are present in Fig. 7.

4.2 Parameter optimization and activation functions

The activation function used in each learning algorithm has its own unique characteristics, and the optimized activation function affects the learning performance. Therefore, the optimal activation function should be selected. In our previous study (Choi 2018), we carried out an experimental investigation of the RMSE so that we could determine the appropriate activation function for each algorithm and

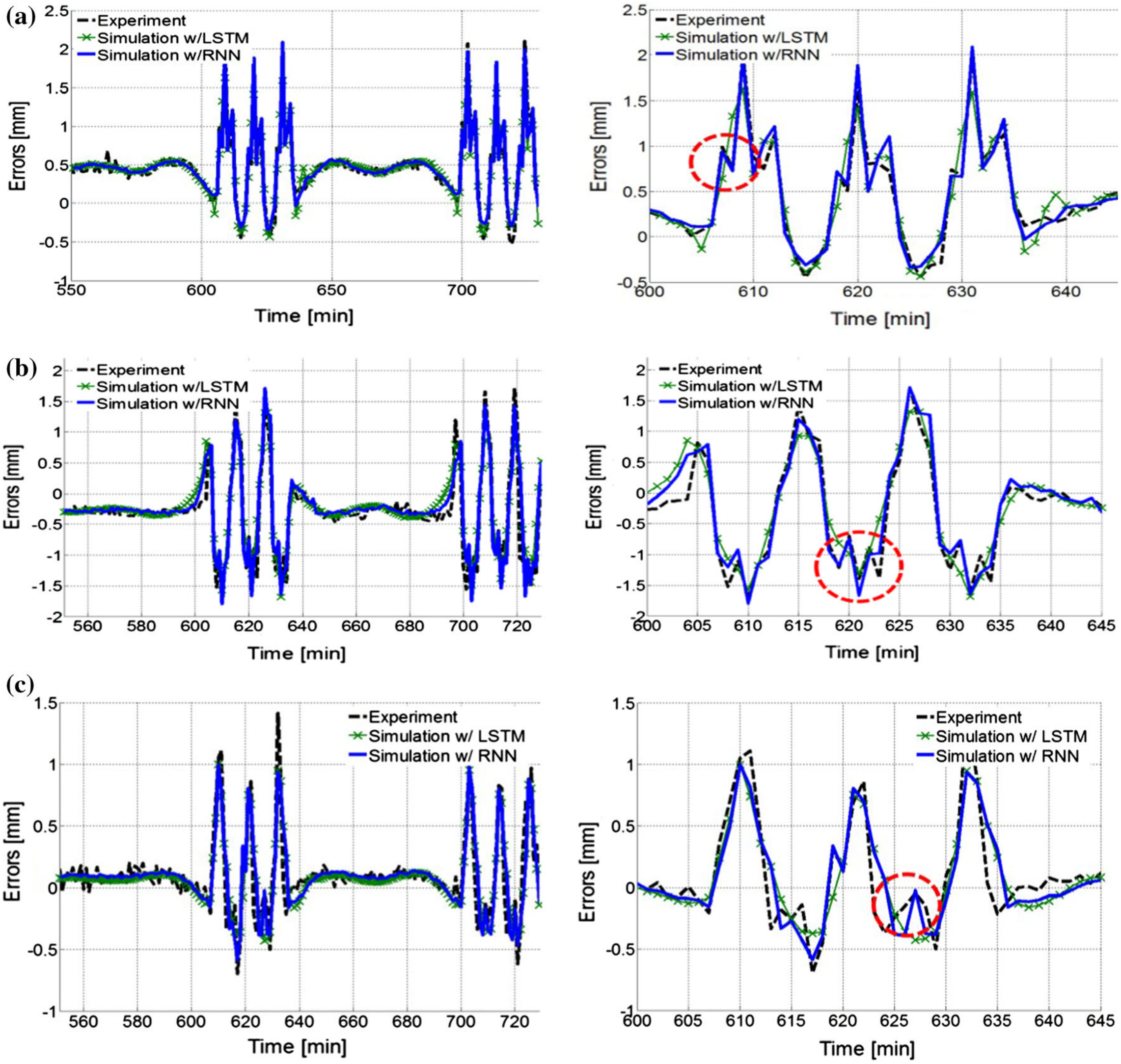


Fig. 10 Learning results for the high frequency components based on a cut-off frequency of 0.0005 Hz: **a** x-axis, **b** y-axis, **c** z-axis

ensure an optimal simulation. We used an LSTM to analyze the low-frequency (non-operating) data and an RNN to analyze the high-frequency (operating) data. When the softsign function as an activation function was used, the RMSE of the low-frequency data at cut-off frequencies of 0.0005 Hz and 0.001 Hz was lowest at 0.1314 mm and 0.0910 mm. When the tanh function was used, the RMSE of the high-frequency data at cut-off frequencies of 0.0005 Hz and 0.001 Hz was lowest at 0.1306 mm and 0.1380 mm, respectively. Therefore, the softsign function was most effective for the LSTM algorithm, and tanh was most effective in the case of the RNN algorithm. This confirms the results from (Le and Zuidema 2015). For this reason, we applied the H-RNN based on these activation functions.

We evaluated the optimal sequence length and hidden dimension parameters by investigating the optimal parameters with respect to various frequencies. We tested between 10 and 15 parameters, because previous research has revealed that learning algorithms converge after a certain number of parameters have been tested (Bengio et al. 1994). As before, we used the LSTM and the softsign function to analyze the low-frequency (non-operating) domain, and the RNN and the tanh function for the high-frequency (operating) domain. In each case, the sequence length has a fixed value, and the hidden dimension increases gradually, converging after 15. As the frequency decrease, the non-operating frequency components are required, as they contain much of this frequency information. In the operating frequency range, the ratio of high-frequency components to noise is relatively large compared with the non-operating frequency domain. Thus, more hidden dimensions are required in this case. However, if the amount of unnecessary noise increases, over-fitting

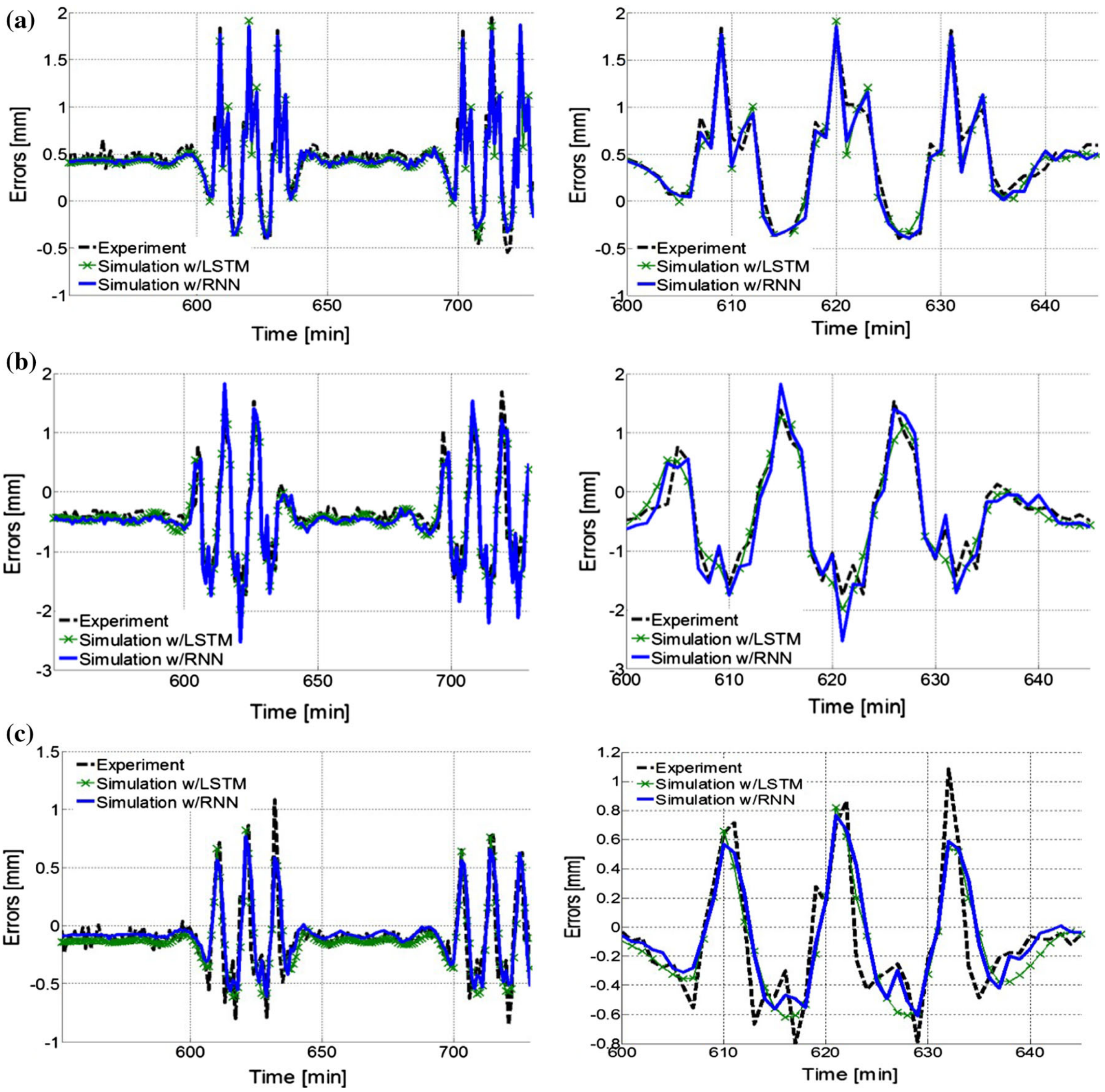


Fig. 11 Learning results for the high frequency components based on a cut-off frequency of 0.001 Hz: **a** *x*-axis, **b** *y*-axis, **c** *z*-axis

Table 3 Root mean square error of the low frequency components (mm)

RMSE(mm)	Axis	LSTM	RNN	RNN/LSTM (%)
0.0005 Hz ∼	*X*	0.184	0.169	109
	Y	0.329	0.241	137
	Z	0.210	0.134	156
0.001 Hz ∼	*X*	0.279	0.256	109
	Y	0.299	0.286	105
	Z	0.151	0.138	110

may occur even if we increase the size of the hidden layer. Therefore, the hidden dimensions tend to converge gradually. We applied this method to determine the optimal parameters for each axis, and each value is specified in Table 1.

4.3 Learning process for low- and high-frequency data using the RNN and LSTM

Figures 8 and 9 show the results of our simulations for the low-frequency (non-operating condition) components with different cut-offs. Figure 8 shows the results in case of the cut-off of 0.0005 Hz, and Fig. 9 shows the case with a cut-off of 0.001 Hz, for the (a) *x*-axis, (b) *y*-axis, (c) *z*-axis. The black dotted line indicates the experimental data, and the blue pointed line and green bold line are the results of the simulations with the RNN and LSTM, respectively. The LSTM had a smaller RMSE than the RNN when the cut-off was 0.0005 Hz (see Table 2). The RNN can also be used in this case, but is less accurate because it contains only short sequence information. We observed a similar tendency when the cut-off was set to 0.001 Hz. However, the RNN

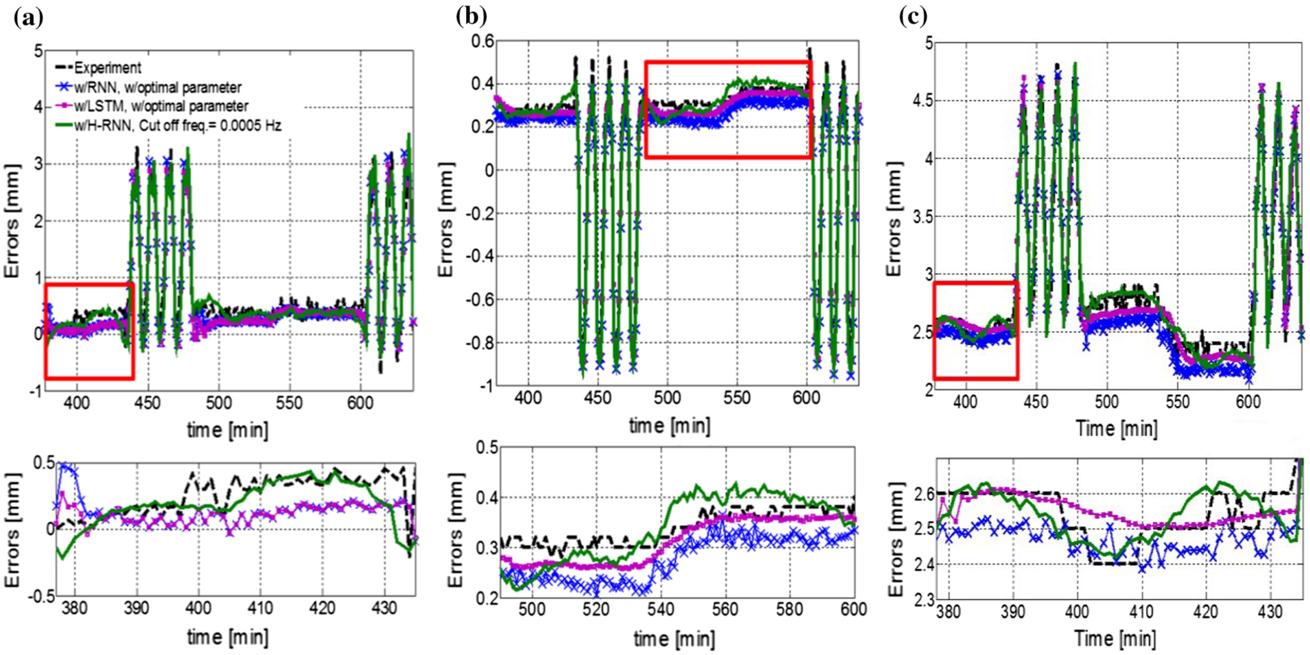


Fig. 12 Learning results of the H-RNN and other algorithms for solving the integrated case: **a** x-axis, **b** y-axis, **c** z-axis

Table 4 Root mean squared error of each algorithm for the entire test set (mm)

Axis	Optimal LSTM	Optimal RNN	H-RNN(0.0005) Hz	Improvement (%)
x	0.242	0.244	0.207	16.9
y	0.060	0.077	0.055	9.0
z	0.164	0.172	0.155	5.8

had a lower RMSE than the LSTM along the y-axis. This is because the data contains more high frequency points as the cut-off frequency increases.

Figures 10 and 11 show the results of the simulated cut-offs of high-frequency (operating) components. Figures 10 and 11 show the results when the cut-off was 0.0005 Hz and 0.001 Hz, for the (a) x-axis, (b) y-axis, (c) z-axis, respectively. Each of the upper graphs are the simulation results from the entire training set, and the bottom graphs are the enlarged test sets obtained while operating the CDPR. The black dotted line represents the experimental data, and the blue bold line and the green pointed line are the results obtained from the RNN and LSTM, respectively. The LSTM converged monotonically, and had a higher RMSE than the RNN (Table 3). In particular, the RNN performed better when the data were changing rapidly, such as in the case of the red circled line. This was because the RNN learns the features of a sudden event easily. This tendency remained the same regardless of the cut-off frequency. Hence, we confirmed experimentally that the proposed learning algorithm is highly accurate.

4.4 Investigation of H-RNN performance

Figure 12 presents the results of the H-RNN simulation for each dataset. The black line is the real experimental test set, while the blue and purple lines show the results of the simulations using the RNN and LSTM, respectively. The green line shows the results from the H-RNN simulation. The RMSE obtained using the H-RNN algorithm was reduced by more than 15%, compared to the maximum error. It can also predict small errors and changes in the nonoperational state, as shown at the bottom of Fig. 12. According to our experiments, the CDPR system has an error of up to 4.7 mm when driven in a cubic trajectory. Also, the hysteresis, creep, and recovery induced by the interaction between loading and unloading actions cause errors to occur while operating the CDPR system. The proposed H-RNN exhibits high pose accuracy (Table 4).

5 Conclusions

We have proposed a novel neural network algorithm to predict the pose estimation error of a CDPR. This error is caused by the non-linear characteristics of the cable. We verified that the frequency-based H-RNN is feasible by constructing a fully constrained CDPR system and operating it with a 3D trajectory so that we could observe the nonlinear characteristics of the cable. We used 729 data, of which 70% were used for the training process and 30% were used for the test process. We used an RNN and an LSTM to learn the features of the data sequences, such as the operation of the end effector over time. CDPRs have high-frequency errors while operating and low-frequency errors when they are not being operated. Therefore, in this study, we constructed an H-RNN algorithm that learns the operating frequency and the non-driving frequency components separately. We separated these frequencies by determining an appropriate cut-off frequency. We obtained this value by identifying the anti-resonance frequency with respect to both the driving frequency and the non-driving frequency. We then confirmed that the H-RNN was more accurate than the RNN or LSTM alone. We also confirmed that determining the cut-off frequency based on the non-operating frequency yields more accurate results because the LSTM has a wider effective frequency range. These results made it possible to predict position errors of CDPRs with high accuracy, in which error varies under both while operating and no operation conditions. The H-RNN algorithm enables us to control the position of the CDPR more accurately. The H-RNN has a lower RMSE than both the optimal RNN and the optimal LSTM, so it is effective for controlling systems that have errors across a range of frequencies.

Acknowledgements This research was supported by Development of Space Core Technology Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (2017M1A3A3A02016340) and from Human Resource Development of the Korea Institute of Energy Technology Evaluation and Planning (KETEP) and the Ministry of Trade, Industry and Energy (MOTIE) of the Republic of Korea (20174030201530).

References

Anderson CW (1989) Learning to control an inverted pendulum using neural networks. IEEE Control Syst Mag 9:31–37

Bengio Y, Simard P, Frasconi P (1994) Learning long-term dependencies with gradient descent is difficult. IEEE Trans Neural Netw 5(2):157–166

Bengio Y, Boulanger-Lewandowski N, Pascanu R (2013) Advances in optimizing recurrent networks. In: IEEE international conference acoustics, speech and signal processing (ICASSP), pp 8624–8628

Chattopadhyay R (1997) Textile rope—a review. Indian J Fibre Text Res 11:360–368

Choi SH (2018) Frequency—based hybrid recurrent neural network for improving the pose accuracy of cable driven parallel robot. The degree of Master thesis, University of Gachon

Choi SH, Park KS (2018) Integrated and nonlinear dynamic model of a polymer Cable for low-speed cable-driven parallel robots. Microsyst Technol 24(11):4677–4687

Gobbi M, Mastinu G, Previati G (2011) A method for measuring the inertia properties of rigid bodies. Mech Syst Signal Process 25:305–308

Hannaford B, Rosen J, Friedman DW, King H, Roan P, Cheng L, Glozman D, Ma J, Kosari SN, White L (2013) Raven-II: an open platform for surgical robotics research. IEEE Trans Biomed Eng 60:954–959

Hyun Dong D (2017) Vibration-based tension estimation of steel wire ropes under the low axial tension in cable driven parallel robot. The degree of Master thesis, University of Gachon

Jung S, IEEE Member, Kim SS (2008) Control experiment of a wheel-driven mobile inverted pendulum using neural network. IEEE Trans Control Syst Technol 16(2):297–303

Le P, Zuidema W (2015) Compositional distributional semantics with long short term memory. arXiv preprint arXiv:1503.02510

Levin AU, Narendra KS (1996) Control of nonlinear dynamical systems using neural networks—part 2: observability, identification, and control. IEEE Trans Neural Netw 7(1):30–42

Li CD, Yi JQ, Yu Y, Zhao DB (2010) Inverse control of cable-driven parallel mechanism using type-2 fuzzy neural network. Acta Autom Sin 36(3):459–464

Merlet JP (2009) Analysis of wire elasticity for wire-driven parallel robots. In: Proceedings of EUCOMES 8, pp 471–478

Merlet JP, Daney D (2010) A portable, modular parallel wire crane for rescue operations. In: Robotics and automation (ICRA) and IEEE international conference, pp 2834–2839

Miyasaka M, Haghighipanah M, Li Y, Hannaford B (2016) Hysteresis model of longitudinally loaded cable for cable driven robots and identification of the parameters. In: IEEE international conference on robotics and automation (ICRA), pp 4015–4057

Pott A (2012) Influence of pulley kinematics on cable-driven parallel robots. In: Lenarcic J (ed) Latest advances in robot kinematics. Springer Netherland, Dordrecht, pp 197–204

Pott A (2017) Increase of position accuracy for cable-driven parallel robots using a model for elongation of plastic fiber ropes. New Trends Mech Mach Sci 43:335343

Wang D, Liu D, Li H, Ma H (2014) Neural-network-based robust optimal control design for a class of uncertain nonlinear systems via adaptive dynamic programming. Inf Sci 282:167–179

Williams II RL, Xin M, Bosscher P (2008) Contour-crafting-cartesian-cable robot system concepts: workspace and Stiffness comparisons. In: ASME 2008 international design engineering technical conferences and computers and information in engineering conference, pp 3–6

Yan Z, Wang J (2014) Robust model predictive control of nonlinear systems with un modeled dynamics and bounded uncertainties based on neural networks. IEEE Trans Neural Netw Learn Syst 25(3):457–469