Tests of Inflatable Structure Shape Control Using Genetic Algorithm and Neural Network

Fujun Peng*, Yan-Ru Hu†, Alfred Ng‡
Canadian Space Agency, 6767 Route de L'Aeroport, St.-Hubert, QC, J3Y 8Y9, Canada

Inflatable space structures need to maintain in a desired shape in space in order to achieve satisfactory performance. The active shape control technique has shown its advantages in solving this problem. Due to strong non-linear properties of the inflatable structures, it is a challenging task to model the inflatable structure properties and to find optimal control output. In this paper, a scheme is proposed based on genetic algorithm and neural network, which is then verified on the shape control of a small size membrane structure. The membrane to be controlled is a 200mm × 300mm rectangular Kapton membrane, pulled by two tensions along each edge. Different combinations of the tensions produce various wrinkles on the membrane. A neural network model is developed to map boundary stretching tensions and space environment to membrane flatness, and then is used to estimate the membrane flatness. The genetic algorithm is utilized to search the best tension combinations from the neural network model to minimize the membrane wrinkle amplitude. An active control system is developed and tests are performed. The results show that genetic algorithm works very well in optimizing the tensions and neural network is effective to estimate the flatness of the membrane.

Nomenclature

x_i	=	neural network input
v_{ij} , w_{jk}	=	neural network weights
β_j , ρ_k	=	neural network thresholds
Y_k	=	neural network output
γ_j	=	combining function output
z_j	=	activation function output
n	=	number of neurons in input layer
p	=	number of neurons in hidden layer
m	=	number of neurons in output layer

I. Introduction

INFLATABLE structures have attracted much interest in the space community due to their unique advantages in achieving low mass and high packaging efficiency.[1,2] Their ultra-lightweight and small-volume properties in turn can potentially reduce the overall space program cost by reducing the launch vehicle size requirement. Inflatable structures can also reduce total system mass and deployment system complexity, thereby increasing system reliability. This type of structures has been envisioned for many space applications such as large telescopes, antennas, solar sails, sun shields, solar arrays, etc.[1-4].

We are currently working on an in-house R&D project in the development of a large surface area to mass ratio inflatable space structure with possible applications as a Synthetic Aperture Radar (SAR) antenna and solar array. The key components of this inflatable structure are inflatable tubes, membrane and the links installed in-between stretching the membrane (see Fig.1). It can be rolled into a small volume and fixed on a small satellite bus for launching. When it gets into orbit, the inflatable tubes are filled with gas and roll out, and the Kapton membrane will be deployed accordingly.

* Corresponding author, NSERC Research Fellow, Spacecraft Engineering Directorate.
† Research Scientist, Control & Analysis Group, Spacecraft Engineering Directorate.
‡ Manager, Control & Analysis Group, Spacecraft Engineering Directorate.

It is expected that the membrane will be subjected to flatness problem during its lifetime in orbit due to the thermal variation in space. A pure passive control method may not be sufficient to maintain the membrane flatness. Hence an active control system is proposed to adjust the tensions according to the thermal variation. The genetic algorithm is applied to search for the optimal tensions that minimize the membrane wrinkles. Actuators will be installed in series with the links, such that the tensions stretching the membrane can be exerted. To predict the membrane flatness in space where direct measurement of membrane flatness is difficult, a neural network model is proposed to map boundary stretching tensions and space environment to membrane flatness. After neural network training is completed, the membrane flatness can be estimated by inputting the measured stretching tensions and space environment to the neural network model.

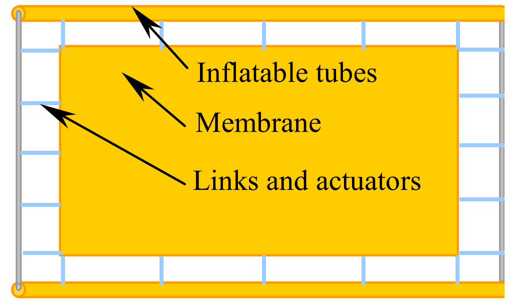

Figure 1. Sketch of the inflatable structure

This paper presents some experimental results obtained in controlling the flatness of a rectangular membrane. The membrane to be controlled is a 200mm × 300mm rectangular Kapton membrane, pulled by two tensions along each edge. The active control system is based on genetic algorithm and neural network technique. Neural network model maps boundary stretching tensions and space environment to membrane flatness, and then is used to evaluate the membrane flatness. The genetic algorithm searches for the best tension combinations from the neural network model to minimize the membrane wrinkle amplitude. Experimental results show that genetic algorithm can find the optimal tension very quickly and neural network is effective to estimate the flatness of the membrane.

II. Genetic Algorithm

The genetic algorithm is an optimization searching technique derived from the mechanics of natural selection and genetics. This mechanism has been mathematically shown to eventually "converge" to the best possible solution. Compared to traditional search and optimization procedures, the genetic algorithm is robust, and generally more straightforward to use. It is stochastic in nature, thus is capable of searching the entire solution space with more likelihood of finding the global optimum. The genetic algorithm is applicable to both linear and nonlinear systems where little or no *a priori* knowledge of the system is given.[5,6]

To implement the search for the optimal tensions for the membrane, all the parameters (here, they are the amplitudes of tensions) to be optimized are first mapped (coded) into a chromosome, each parameter corresponding to one particular portion of the chromosome, then the following steps are executed to search for the best solutions (see Fig. 2):

1. **[Initialization]** Randomly generate the initial population (possible tension combinations)
2. **[Fitness]** Evaluate the fitness of each individual in the population (smaller amplitude means better fit)
3. **[Criteria met?]** Check if the end condition is satisfied: if yes, stop and return the best solution; if no, generate new population for a repeat execution of the algorithm
4. **[New population]** Create a new population by repeating the following steps:
    - **[Selection]** Select two parent individuals based on their fitness (the better the fitness, the bigger chance to be selected).
    - **[Crossover]** Using a crossover

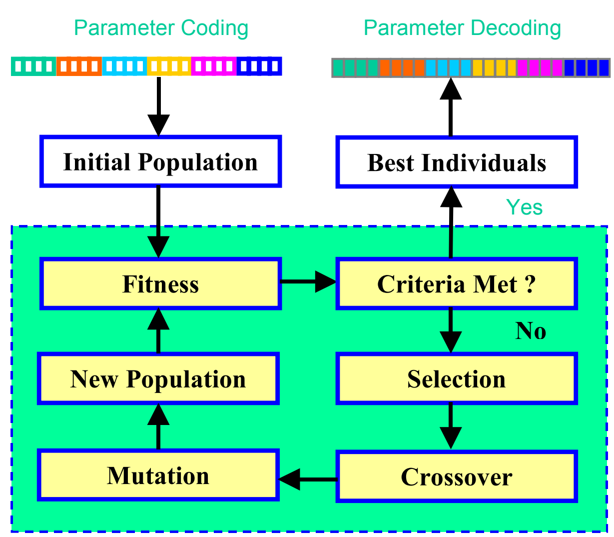

Figure 2. Block diagram of genetic algorithm

probability, cross over the parent individuals to form a new offspring.

- **[Mutation]** Using a mutation probability, mutate new offspring at each locus (position in an individual).
- **[Accepting]** Place new offspring in a new population.

5. **[Loop]** Go to step **2**

After the best individuals are obtained, decode them to the required parameters, and choose one as the optimized solution.

III. Neural Network

A neural network is a system composed of single or multiple layers of processing elements operating in parallel, whose function is determined by network structure, connection strengths, and the processing performed at computing elements or nodes. Neural network topologies can be structured to generate arbitrarily complex decision regions for stimulus-response pairs, hence they are perfectly suited for mapping input-output relationships. It is proven that a multi-layer neural network can approximate any continuous nonlinear function arbitrarily well on a compact interval, provided sufficient hidden neurons are available. Its generalization ability makes it intelligent with respect to fresh and unknown data. Neural networks have found wide range of applications in image processing, pattern recognition, speech production, nonlinear modeling, control and robotics and optimization, etc. [7-9] Figure 3 shows a depiction of feedforward neural network.

To estimate membrane flatness, we need to map space environment and boundary tensions to membrane flatness using neural network, so that membrane flatness could be estimated as long as the environment and tensions are obtained. Such a mapping relationship can be treated as a static problem, therefore a static feedforward neural network is a good choice. Here we use one hidden layer MLP (Multi-layer Perceptron), which has been proven to be able to approximate any continuous non-linear function to an arbitrary degree of accuracy if sufficient hidden layer neurons are used. [10,11] The structure of the MLP network is depicted in Fig. 3.

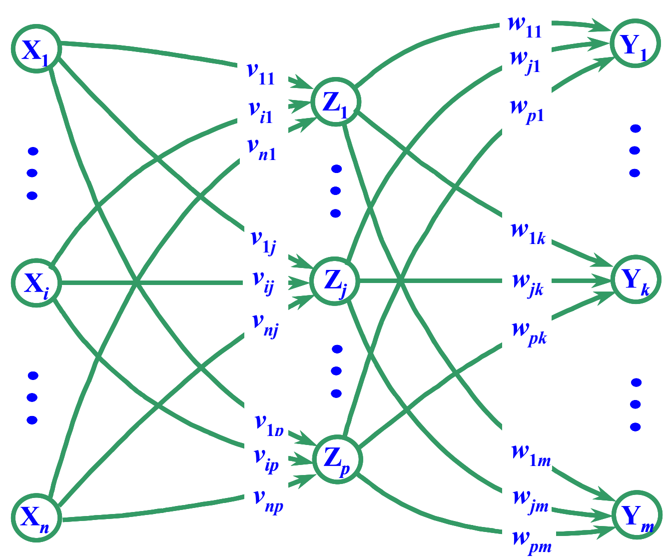

Figure 3. Depiction of feedforward neural networks

The basic operation of a hidden layer neuron involves performing two functions: combining function and activation function (logistic sigmoid is used here). This is illustrated in Fig. 4.

$$\gamma_j = \sum_{i=1}^{n} v_{ij} x_i \qquad (1)$$

$$z_j = 1 / (1 + \exp(-\gamma_j - \beta_j)),\ j = 1, \cdots, p \qquad (2)$$

where n, p and β_j are the number of neurons in the input layer, the number of neurons in the hidden layer and thresholds, respectively. Similarly, the output of the k-th neuron in the output layer can be written as:

$$Y_k = f_{output}\left(\sum_{j=1}^{p} w_{jk} z_j + \rho_k\right),\ k = 1, \cdots, m \qquad (3)$$

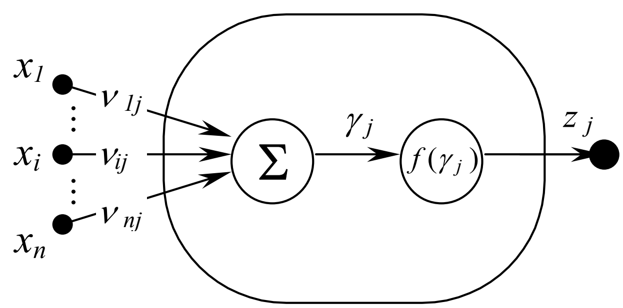

Figure 4. Basic operation of an artificial neuron

where m and ρ_k are the number of neurons in the output layer and thresholds, respectively.

The weights v_{ij}, w_{jk} and thresholds β_j, ρ_k are unknown parameters and should be determined by a supervised training. The block diagram of the supervised training is shown in Fig. 5. The same environment and tension data are input to the real inflatable structure and the neural network, and a training algorithm adjusts neural network parameters to minimize the error of their outputs. After the training completes, the MLP network can be used to estimate the membrane flatness. Provide the measured environment data and boundary tensions into the input vector [x_1, x_2, … x_n], the membrane flatness can be estimated using Eqs. (1) to (3).

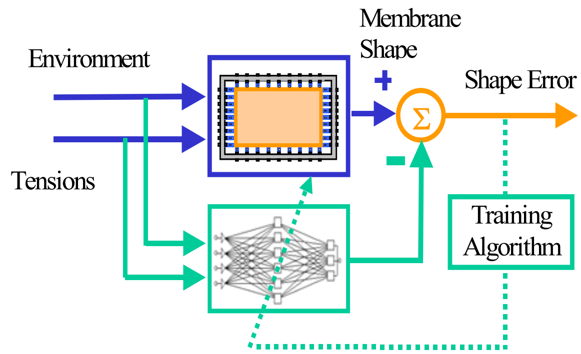

Figure 5. Block diagram of neural network supervised training

IV. Experimental Setup and Control System Implementation

The structure to be used for the neural network validation is shown in Fig. 6. The membrane is a 200mm×300mm rectangular Kapton Membrane, which is stressed by 8 discrete links installed between the membrane boundaries and the aluminum frame. A local thermal load source is placed under the membrane (not visible in Fig. 6). The membrane flatness is dependent on the local thermal load and the tension combinations. In order to obtain different tension combinations for the neural network training, four shape memory alloy wire actuators (0.25mm in diameter and 100mm in length) are installed on links 1 to 4. To acquire the values of tensions, strain gages are glued onto small and thin aluminum strips, which are then installed on links 1 to 4. These tension measurement elements are calibrated using a load cell before tests are performed. The arrangement of the SMA actuator, the strain gage and links is sketched in Fig. 7. To avoid symmetric properties, two springs are intentionally installed on links 5 and 8, whereas links 6 and 7 are rigid.

To exert different training tensions onto the membrnane, more work needs to be done on the SMA actuators. The SMA wire actuator cannot be used directly for exerting tensions due to its poor stability and controllability. It is hard to identify a reliable relationship between strain output and electrical input current, and a steady strain output is even not achievable by applying a fixed current. So a feedback SMA controller has been developed to ensure the required tension could be achieved with good accuracy. [12]

A vision system is used to measure the membrane flatness (see Fig. 8). A digital camera and a light projector are used. The camera is calibrated in 3D world space, which allows a pixel in the camera plane to be mapped to a line (cone) radiating from the focal point in the world

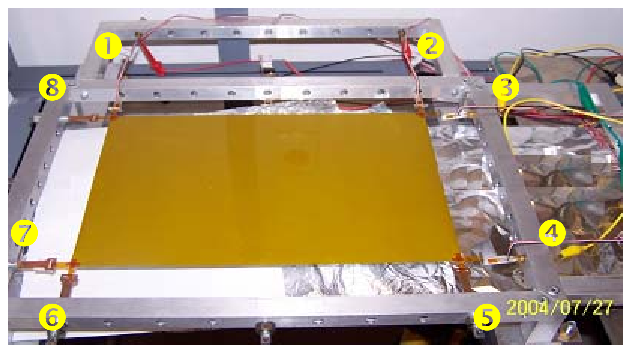

Figure 6. Membrane structure used for neural network validation

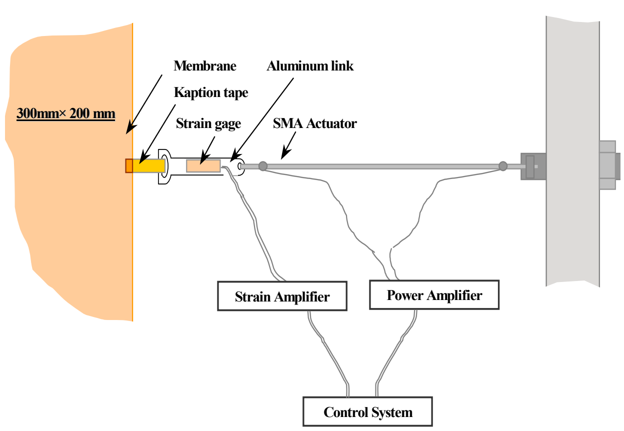

Figure 7. Arrangement of actuators and sensors

coordinate system. The light projector projects multiple light planes onto the membrane surface, which produce curves on the membrane. These planes are also calibrated in the world coordinate space. The camera observes the points on curves projected by projector. Locations of these points are thus determined by the light plane and the associated radiating line from the camera focal point. Their 3D coordinates can be calculated by solving the equations of the light plane and the radiating line. The membrane flatness is defined as the standard deviation of the obtained 3D coordinates. In order for the vision system camera to see the curves clearly, a very thin coating is put on one side of the membrane. Test results show that the vision system works very well. It takes only 0.1s to obtain 330 points of 3D coordinates, with the accuracy on the order of 0.1mm.[13] The membrane flatness data obtained from this vision system will be used for network training. When the control system is performing control, the vision system will not feedback the measured data to the control system. Instead, it just monitors the control effectiveness.

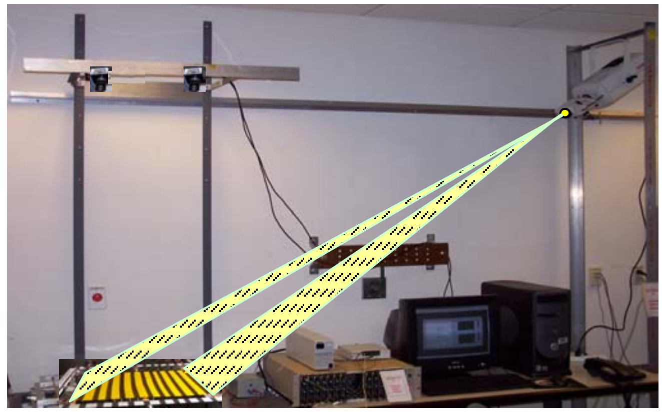

Figure 8. Vision system for membrane flatness measurement

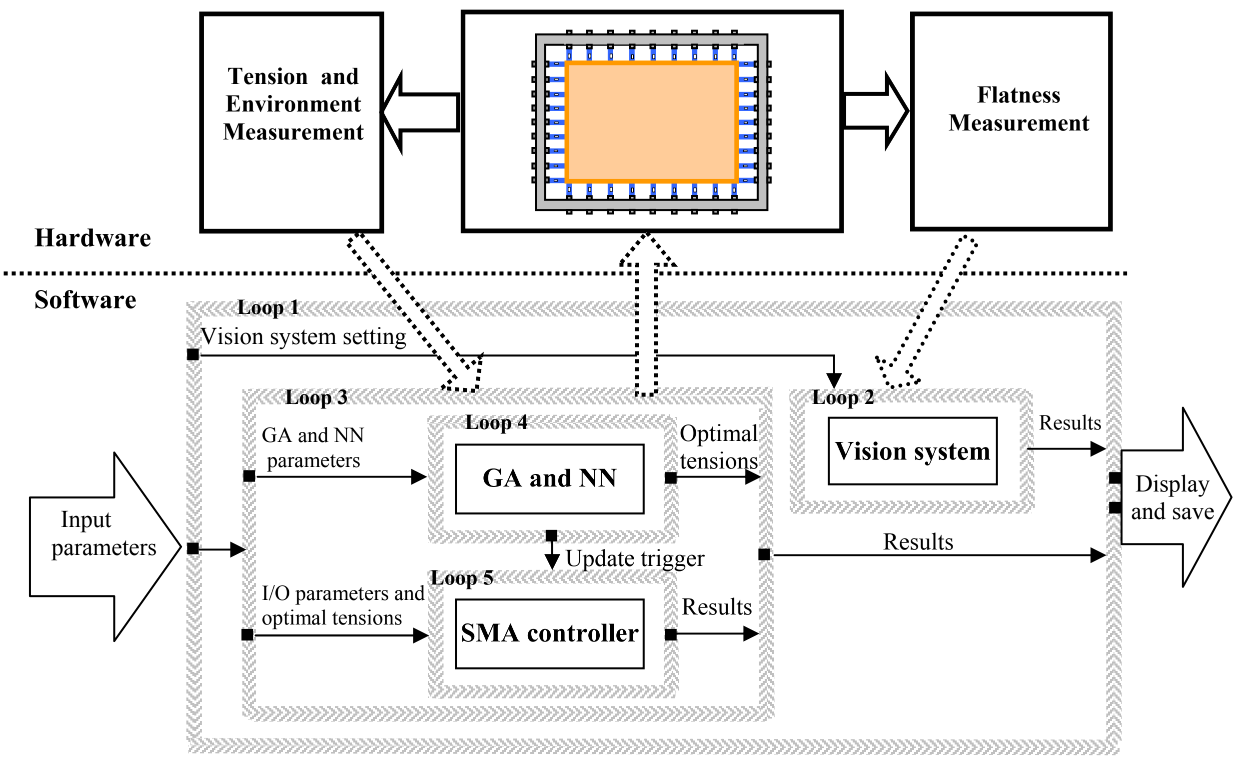

Figure 9. Block diagram of the control system implementation

The entire control system consists of three modules, which are organized in five loops, as illustrated in Fig. 9. Loop 1 receives parameters from GUI (Graphical User Interface), and then triggers Loop 2 and Loop 3 with proper parameters. Loop 2 executes the vision system independently with a high rate until it is stopped by a trigger. As mentioned above, the vision system is only used here to monitor the membrane flatness, and the obtained result is compared with the estimated values to evaluate the effectiveness of the proposed GA-NN scheme. Loop 3 triggers the executions of GA-NN module and SMA controller. Most of the time, GA-NN module is working in monitoring mode, i.e., it monitors the environment data in real time and estimates the membrane flatness using current environment data and tensions. If the estimated membrane flatness is acceptable, it will remain in this mode. However, if the environment changes too much and the estimated value shows the membrane is not flat enough, GA-NN module will automatically switch to optimization mode, i.e., it searches for a new optimal tension combination that can re-establish a flat membrane state. After the new optimal tension combination is obtained, GA-NN module will send it to Loop 3 and at the same time trigger the SMA controller to stop, and then GA-NN terminates automatically. After the SMA controller stops, Loop 3 will right away go to the next cycle and transfer the new optimal tension combination to the SMA controller. The control system is implemented using LabView, Matlab and Automation Manager. LabView code controls parameter input, results display and saving and data I/O, including tension measurement and control signal output. Matlab code implements genetic algorithm and neural network. Automation Manager realizes the vision system. LabView code is the master, which coordinates the whole system functioning by using ActiveX technique.

V. Neural Network Training

In order for the neural network to better approximate the mapping from environment temperature and tension combinations to the membrane flatness, the input data (here are the environment temperature and tension combinations) chosen for the neural network training should spread out in a data space, in which a number of optimal tension combinations should exist. After performing preliminary tests, we limit the values of training tensions between 1.32N and 2.64N, and totally 9 values are uniformly chosen in this region. Here we only record the values of tensions applied by the actuators 1 to 4 (denoted as T_1 to T_4 below), and the 9 values of them are: 1.32N, 1.49N, 1.65N, 1.82N, 1.99N, 2.15N, 2.31N, 2.48N, and 2.64N. The total number of tension combinations chosen for neural network training is 9×9×9×9=6561. As for the membrane environment temperature, a small heater is placed under the membrane as a local thermal source, which makes the temperature distribution on the membrane neither uniform nor symmetric. Also, the local thermal load applied to the membrane produces dynamic wrinkles within an area over the heater head, which is fluctuating all the time. For the simplicity consideration, the position of the heater is fixed and its temperature remains unchanged at around 200°C.

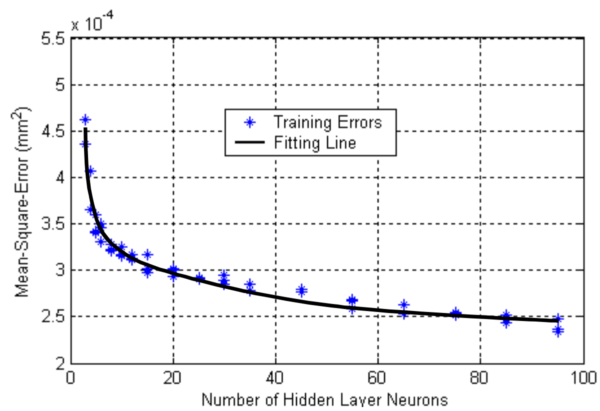

Fig. 10. Training errorsTension and Environment Mesurement

The training tension combinations are exerted to the membrane structure in the following steps: Keep T_1, T_2 and T_3 unchanged at 1.32N and exert the 9 values of T_4 one by one from 1.32N to 2.64N. The corresponding membrane flatness is also recorded one by one by the vision system. Then move T_3 to 1.49N and exert the 9 values of T_4 one by one again from 1.32N to 2.64N (T_1 and T_2 remain unchanged at 1.32N). Repeat this procedure: move T_3 to the next value when finish a cycle for T_4; move T_2 to the next value when finish a cycle for T_3; move T_1 to the next value when finish a cycle for T_2. After this procedure is repeated for 6561 times, all the training tension combinations and the corresponding membrane flatness are obtained and ready for the neural network training. After training data are obtained, 1000 randomly generated tension combinations are exerted to the membrane structure, and corresponding values of membrane flatness are recorded. These data will be used as testing data to validate the trained neural network.

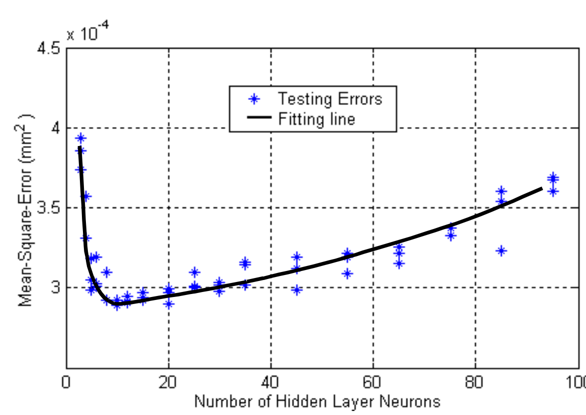

Fig. 11. Testing errors under different hidden layer neurons

As mentioned before, the neural network used here is a MLP network with one hidden layer having

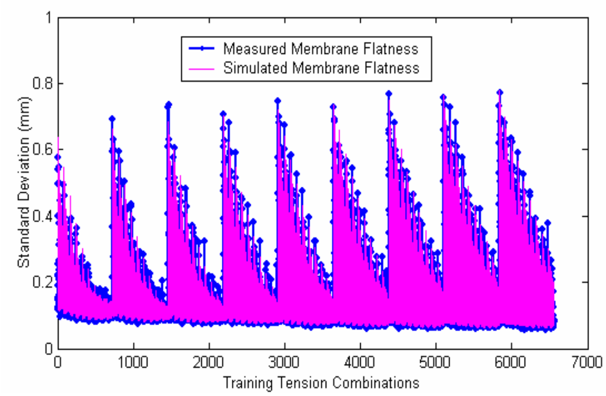

Fig. 12. Measured and simulated membrane flatness corresponding to all training tension combinations

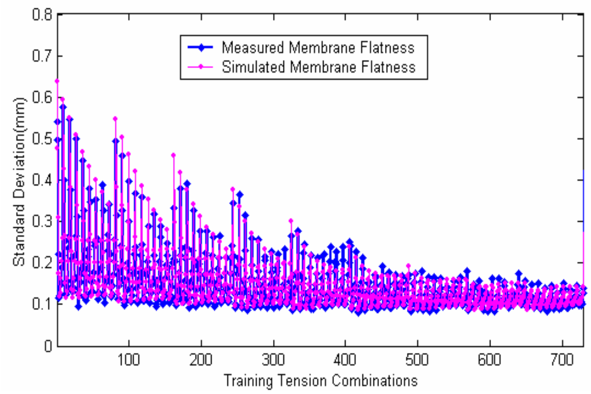

Fig. 13 Measured and simulated membrane flatness in the case that T_1=1.32N

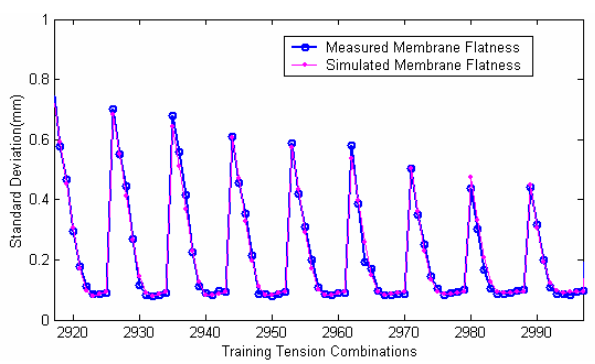

Fig. 14 Measured and simulated membrane flatness in the case that T_1=1.98N and T_2=1.32N

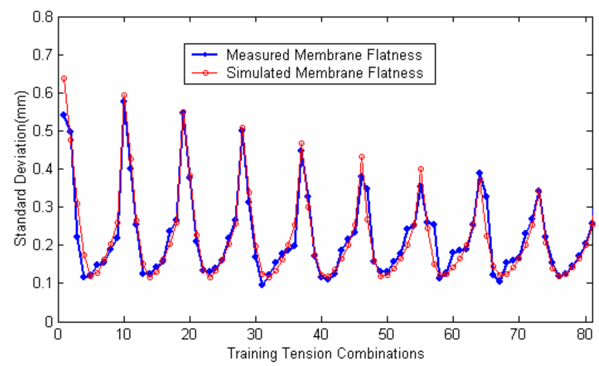

Fig. 15 Measured and simulated membrane flatness in the case that T_1=1.32N and T_2=1.32N

"logistic sigmoid" neurons. It has five inputs, one for environment temperature and the other four for tensions. In the output layer, there is only one neuron with a linear activation function. This only neuron corresponds to the membrane flatness. In order to obtain a good performance of neural network, different numbers of hidden layer neurons are tested. Each case is performed three times and each time the neural network is trained for 300 epochs, and the testing data are then input into the obtained neural network for performance test. The training errors and testing errors are shown in Figs. 10 and 11. We can see that the training error is always descending along with the increase of hidden layer neuron numbers. Whereas for the testing error, it gets the minimal value when the hidden layer neurons are around 10, and then goes up gradually along with the increase of hidden layer neuron numbers. Therefore, the optimal number of hidden layer neurons should be near 10. Fix the hidden layer neuron number at 10, perform network training for 20 times with different training epochs. The obtained minimum testing MSE (mean square error) is 2.8864×10^{-4} mm^2, and the corresponding training error is 3.18744×10^{-4} mm^2. Some of the results are shown in Figs. 12 to 16.

In Fig. 12, the measured membrane flatness (produced by all training tension combinations) and the simulated membrane flatness (obtained by inputting training tension combinations into the neural network trained) are given in comparison. We can see their profiles agree with each other quite well. Similar results can be found in Figs. 13 to 15. This implies that the obtained neural network offers a good simulation to the ways that the four tensions affect the membrane flatness. In Fig. 15, we can see that the simulated membrane flatness is more regular than the measured values. We know that the irregular phenomenon in the measured membrane flatness is mainly caused by the fluctuation of the thermal load induced dynamic wrinkles. Therefore, the regularity of the simulated membrane flatness implies that the optimal neural network obtained tends to eliminate the effect of the dynamic wrinkle fluctuations. This does not mean the neural network frees itself completely from the thermal load. The latter does influence the neural network parameters through its average effect, i.e., the optimal neural network only takes into account the average value of the dynamic wrinkle fluctuations, instead of its instant values. Therefore, when the thermal load dynamic wrinkles contribute much to the membrane bumpiness, the simulated membrane flatness has larger difference from the actual measured values. However, in the case that a good flatness is obtained, the simulated membrane flatness agrees better with the measured values, since the fluctuation magnitude of dynamic wrinkles is very small at this time. This effect can be seen clearly in Figs. 14 and 15.

Figure 16 shows the estimated and measured values of membrane flatness produced by the randomly generated testing tension combinations, instead of training data set. Basically the estimated values agree with the measured values quite well. When the membrane flatness is poor, the estimated errors are larger. In the cases of good flatness, however, the estimated flatness is very close to measured values. This characteristic is quite similar to the training cases. It is worthwhile to point out that this characteristic may not cause any difficulty to practical application since our concern is only the good flatness cases in which neural network offers good approximations.

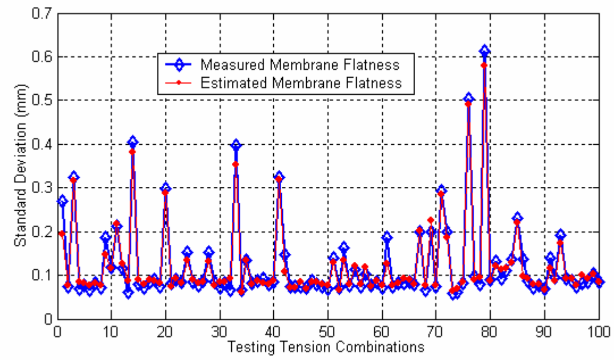

Fig. 16 Measured and estimated membrane flatness corresponding to randomly generated testing data

VI. Test of Membrane Flatness Control

After the optimal trained neural network is obtained, we can perform the test of membrane shape control. First, place a white board at the left side of the set-up and adjust its location such that an image will be reflected on it by the membrane (the light comes from the projector, which is used for projecting lines for the vision system). If the membrane is flat, the lines shined on the membrane will be straight and the reflected image will be straight too. But if there is any wrinkle generated, the distortion will be amplified on the reflected image. Therefore this image can be used as a direct judgment of membrane flatness (see Fig. 17). The initial tensions are all set at 1.32N. For the genetic algorithm, there are 8 individuals in one population, and 4 of them are updated at one time. Other genetic algorithm settings are: gray coding, stochastic universal sampling, multi-point crossover and discrete mutation. The membrane flatness requirement is set at 0.1mm (standard deviation). Turn on the local heater, and when its temperature attains the required 200°C (the same as for the case of neural netork training), start the control system. The genetic algorithm begins to search for the optimal tension combination based on the obtained neural network. When the optimal tension combination is found, it is transferred to the SMA controller and the latter exerts it through SMA actuators. After the real tensions attain the optimal tensions and the vision system completes the recording of the membrane flatness, stop the control system. Figure 18 shows the convergence of the estimated membrane flatness, in which the 15 points marked with circles are the estimated best values of the membrane flatness in the 15 generations of optimization. Figure 19 shows the two images before and after control, by which we can directly see the membrane flatness is improved significantly. To find this best tension combination, totally 15 generations

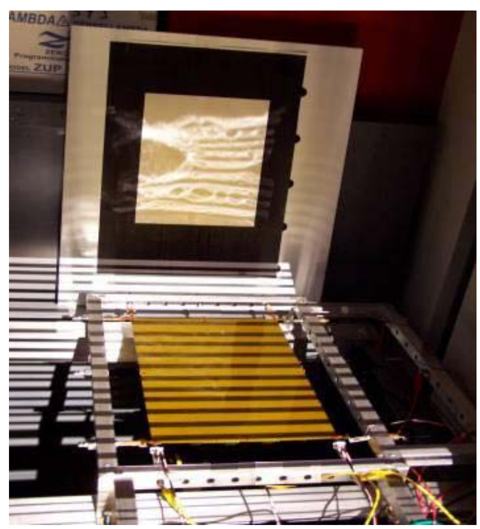

Figure 17. Observation of the membrane flatness

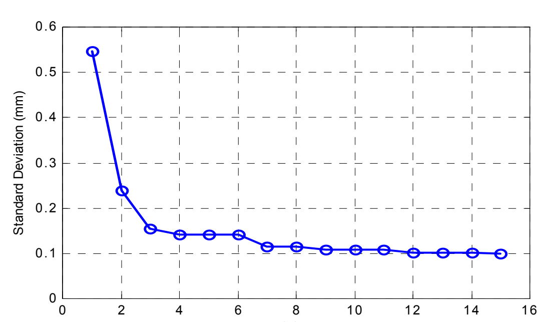

Figure 18. Decrease of estimated membrane flatness

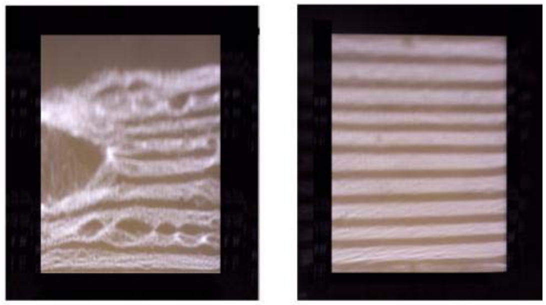

Figure 19. Images before and after control

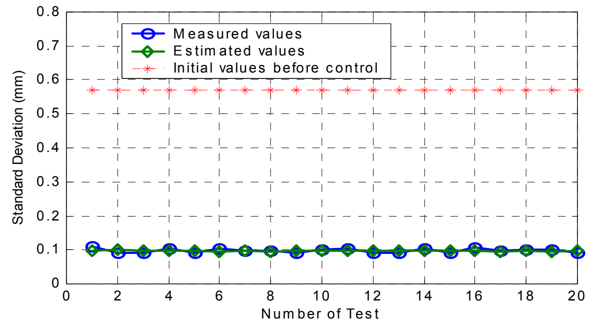

Figure 20. Obtained membrane flatness of the 20 tests

of 64 tension combinations have been tested by the genetic algorithm. This convergence process takes less than one second for our case that the genetic algorithm is coded with Matlab.

Perform the test 19 more times and for each time the membrane flatness requirement remains the same at 0.1mm. The totally 20 times of obtained results of membrane flatness are shown in Fig. 20. We can see that, for all the 20 tests, the genetic algorithm successfully finds the optimal tension combination. Also, the estimated membrane flatness agrees very well with the real measured values, and both are very close to the membrane flatness requirement, thus demonstrates that the neural network scheme is effective in the estimation of the membrane flatness.

VII. Conclusion

This paper presents a genetic algorithm and neural network based scheme for active shape control of inflatable structures. Preliminary experimental tests are performed on a small size membrane structure and the results demonstrate the effectiveness of the proposed neural network scheme in the shape estimation of this type of structures with strong non-linearities. Test results also indicate that the genetic algorithm can find the optimal tension combinations effectively and quickly. More tests are ongoing with varying local thermal loads applied.

References

[1]Jenkins, C. H. M. (ed.), *Gossamer Spacecraft: Membrane and Inflatable Structures Technology for Space Applications*, Progress in Astronautics and Aeronautics, Vol.191, AIAA, 2001

[2]Cadogan, D., and Grahne M., "Inflatable Space Structures: A New Paradagm for Space Structure Design", *Proceedings of the 49th International Astronautical Congress*, Sept.28-Oct 2, 1998, Melbourne, Australia, IAF-98-I.1.02

[3]Lin, J. K. H., and Cadogan, D. P., "An Inflatable Microstrip Reflectarray Concept for Ka-Band Applications", *Proceedings of the 41st AIAA/ASME/ASCE/ AHS/ASC Structures, Structural Dynamics and Materials Conference & Exhibit*, April 3-6, 2000, Atlanta, AIAA2000-1831

[4]Karooka, D. K., and Jensen, D. W., "Advanced Space Structure Concepts And Their Development", *Proceedings of the 42nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference and Exhibit*, April 16-19, 2001, Seattle, AIAA-2001-1257

[5]Hansen, C.H., Simpson, M.T., and Cazzolato, B.S., “Genetic Algorithms for Active Sound and Vibration Control,” *Proceedings of the IEE International On-line Conference on Active Control of Sound and Vibration*, 1999.

[6]Chambers, L.(ed.), *The Practical Handbook of Genetic Algorithms: Applications*, Second Edition, CRC Press, 2000

[7]Fausett, L., *Fundamentals of Neural Networks, Architectures, Algorithm and Applications*, Prentice-Hall, Inc., 1994

[8]Liang, Y. C., et al., "Successive approximation training algorithm for feedforward neural networks", *Neurocomputing*, Vol.42, 2002, pp.311-322.

[9]Suykens, J. A. K., Vandewalle, J. P. L., and Moor, B. L. R., *Artificial neural networks for modeling and control of non-linear systems*, Kluwer Academic Publishers, 1996

[10]Cybenko, G., “Approximations by superposition of a sigmoidal function”, *Mathematics of Control, Signal and Systems*, **2**, 1989, pp.303-314

[11]Funahashi, K., “On the approximate realization of continuous mappings by neural networks”, *Neural Networks*, **2**, 1989, pp.183-192

[12]Peng, F., Jiang, X. X., Hu, Y.-R., and Ng, A., “Application of Shape Memory Alloy Actuators in Active Shape Control of Inflatable Space Structures”’, *Proceedings of the 2005 IEEE Aerospace Conference*, Mar.5-12, 2005, Big Sky, MT, USA

[13]Peng, F., Hu, Y.-R., and Ng, A., “Measurement of membrane flatness for active control of inflatable structures”, *Proceedings of 7th CanSmart International Workshop on Smart Materials & Structures*, Oct. 21-22, 2004, Montreal, Canada