# Accelerated Co-evolutionary Algorithms

**Jong-Han Kim***
MADC, Agency for Defense Development
Yusung P.O.Box 35-3, Taejeon, South Korea

**Min-Jea Tahk****
Division of Aerospace Engineering, KAIST
Kusung-dong, Yusung-gu, Taejeon, South Korea

## Abstract

A new co-evolutionary algorithm, of which the convergence speed is accelerated by neural networks, is proposed and verified in this paper. To reduce computational load required for co-evolutionary optimization processes, the cost function and constraint information is stored in the neural networks, and the extra offspring group, whose cost is computed by the neural networks, is generated. It increases the offspring population size without overloading computational effort; therefore, the convergence speed is accelerated. The proposed algorithm is applied to attitude control design of flexible satellites, and it is verified by computer simulations and experiments using a torque-free air bearing system.

**Key Word** : co-evolutionary algorithm, neural networks

## Introduction

A great number of control techniques for aerospace systems have been proposed. The earlier groups are mainly based on the classical PID design approaches(e.g.[1][2]), while the others employed the advanced modern control approaches(e.g.[3]-[5]). The latter methods provide fine performance, sometimes even better than that of the classical ones. However, many of them are too complicated to go on board and have few flight histories. Generally, aerospace engineers are still hesitating to employ new technologies that aren't flight-proven yet.

Meanwhile, well-designed classical controllers provide fair performance, and are already flight-proven. However, the performance of them mainly depends on the designer's inspiration. To clear up the difficulties, directly optimizing controller gains is often employed[6].

Evolutionary algorithms with proper cost selection automate the gain selection process. It is an appropriate tool for problems with a single cost function and no constraint. However, many of practical design problems fall into constrained optimization problems. Robustness-stability conditions can be formulated as constraints, and physical constraints like actuator limits can be.

Recently developed Co-Evolutionary Augmented Lagrangian Method (CEALM), which is powerful for solving constrained optimization problems, is one of solutions[7]. Once the cost is properly chosen, the Lagrangian function is formulated, and augmented for convexity. Setting and solving given design problem as a minimax game between the controller gain group and the Lagrangian multiplier group, we obtain the optimal controller, which satisfies all the constraints.

* Researcher, MADC, ADD
** Professor, Div. of AE, KAIST
E-mail : mjtahk@fdcl.kaist.ac.kr, TEL : 042-869-3718, FAX : 042-869-3710

The CEALM is suitable for the practical controller design problems that have a few requirements[8].

For general evolutionary optimization techniques, the cost value is frequently computed, resulting in a heavy computational load. In this paper, a new co-evolutionary algorithm, whose convergence is accelerated by neural networks, is proposed. Cost evaluation is the most time-consuming process in practical optimization problems, but conventional evolutionary algorithms use cost information only for fitness evaluation. In the new algorithm, the cost information is not discarded but stored in the neural networks for further use. The neural networks is trained from the early stages of evolution. After sufficient training is achieved, the Extra Offspring group, whose cost is computed by neural networks, is generated. It increases, doubles or triples the offspring population. For most practical problems, the computational load required for neural network training or computation is much smaller than that required for evaluation of the original cost. Therefore, the convergence of the population is accelerated.

In this paper, acceleration methods for co-evolutionary algorithms are proposed. As an application, a single axis attitude control of a flexible space structure is considered. Finally, it is verified by nonlinear computer simulation, and experimentally demonstrated on a torque-free air bearing test facility.

## Accelerated Co-evolutionary Algorithms

### Acceleration Methods

For most practical optimization problems, the cost is determined by complicated functions such as integration or transcendental functions. Moreover, the evolution-based algorithms need cost evaluations very frequently, which results in a heavy computational load and time[9]. Increasing population size is definitely the best way to help the groups converge in the early stages of the evolution, but in that case, every single generation takes much more computation time, and termination time itself doesn't always reduce. The population size is one of the parameter that directly determines the convergence speed. Usually, the population size is experimentally determined for increasing population leads to increased computational load.

The accelerated evolutionary algorithms are motivated from the universal approximation property of the multilayer feedforward neural networks[10]. In the new algorithm, the cost information is reused to train the neural networks. The neural network weights are adapted whenever the cost is evaluated. After the cost evaluation process, the Extra Offspring group whose cost is computed by the neural networks is generated. The number of extra offspring population is to be determined exploratory. The larger extra population size tends to accelerate the convergence more, but can lead itself to wrong evolutionary directions. Hence, proper selection of the size determines the convergence characteristics. A similar size of the normal offspring group's is a modest choice. One might vary the size from generation to generation to improve the convergence speed. After the extra offspring generation, the ordinary offspring group and the extra offspring group proceed to the fitness check step together.

### Accelerated Co-evolutionary Algorithms

Co-evolutionary algorithms are originally developed for solving minimax problems[6][11], and can be applied to constrained optimization problems with Co-evolutionary augmented Lagrangian method[7]. It simulates a competitive evolution of two groups of opposite objectives; one tries to minimize the given cost function while the other tries to maximize. Same rules for reproduction, and mutation of evolutionary algorithms can be adopted for co-evolutionary algorithms.

The core idea of accelerated co-evolutionary algorithms is an improvement of that proposed in [12], with introduction of the extra offspring concept. The neural networks learns cost information and constraint information, and the extra offspring group of the two groups are generated. Each group evolves in the following way, in the accelerated co-evolutionary algorithm(ACEA).

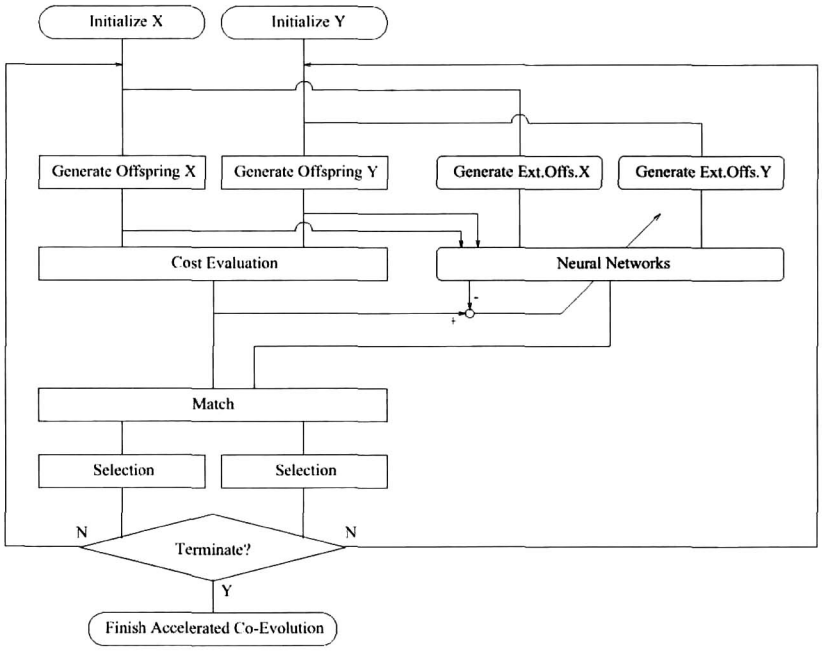



**Fig. 1. Flow chart for Accelerated Co-evolutionary Algorithms**

1. Initialization
2. Offspring generation
3. Cost evaluation *with neural network training*
4. *Extra offspring generation*
5. *Cost evaluation for extra offspring (cost obtained by neural network computation)*
6. Match
7. Selection
8. Termination condition

## A Numerical Example

Ex.1) For the following plant, find the proportional feedback gain that maximizes the closed-loop system agility, with the overshoot less than 20%.

$$\text{Plant Dynamics} : \quad G(s) = \frac{1}{s(s+2)}$$

The design statement is equivalent to the following constrained optimization problem.

$$\text{Minimize} : \quad -\min(|\omega_n|)$$

$$\text{Subject to} : \quad \max(y) \leq 1.2$$

where $\omega_n$ is the natural frequency of the closed-loop system, and $y$ is the output histories.

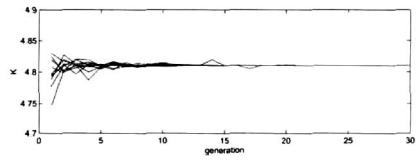
The parameters in Table 1 are employed for the acceleration scheme. The sigmoid activation function is employed for the neural network implementation.

The co-evolution histories are

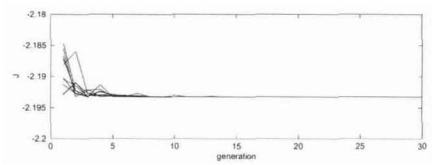
**Table 1. ACEA Parameters for the example**

| | |
|---|---|
| Number of Parents( $N_p$ ) | 20 |
| Number of Offspring( $N_o$ ) | 50 |
| Number of Extra Offspring( $N_{eo}$ ) | 50 |
| Selection Type | ( $\mu$ , $\lambda$ ) |
| Network Type | Single hidden layer/ Sigmoid |
| Number of Hidden Nodes | 10 |

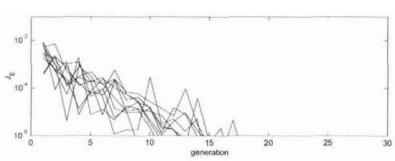
presented in Fig. 2 and Fig. 3. Histories by the accelerated algorithm and the conventional algorithm are provided for comparisons. represents the deviation of the best cost from the optimal cost . The deviation history is appended for fair description of convergence. The proposed algorithm clearly accelerates the convergence. Almost all of the trials in the accelerated algorithm converged in tolerance of within the 15'th generation, while the original algorithm did not.




(a) $K$ histories

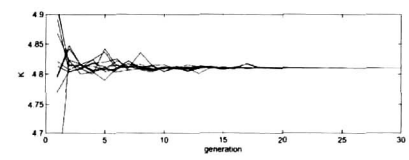



(b) $J$ histories

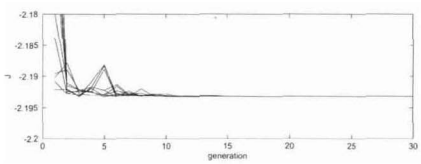



(c) $J_E$ histories

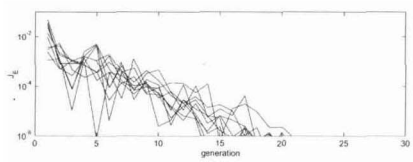
Fig. 2. ACEA histories
( $N_p$:$N_o$:$N_{eo}$ = 20:50:50 )




(a) $K$ histories




(b) $J$ histories




(c) $J_E$ histories

Fig. 3. CEA histories
( $N_p$:$N_o$: = 20:50 )

## Applications

### Flexible Space Structures

Future space missions will need extremely high technology, which are unreal today, and spacecrafts will be equipped with complex electronics and structures. Then, more complicated attitude control is required for considering oscillatory motion caused by the structures[13]. Usually, very stiff materials are selected for structures so that the natural frequency becomes higher. In fact, small external force/torque is exerted on a satellite in the space environment, non-stiff materials are enough. Stiff materials are preferred not only for structural safety, but also for easy attitude control in this case. If attitude control algorithm confidently covers the structural vibration, more flexible and light materials can be used and it enables inexpensive production, and low-cost launch. Some experimental, brave missions employ non-metal flexible structures, wires, thin panels, and membranes, etc[14]. That makes attitude control harder.

In this chapter, a single axis attitude control of flexible space structures is concerned. A flexible satellite is mathematically modeled, and an attitude controller is to be designed by the accelerated co-evolutionary algorithm.

### Dynamic Modeling

The flexible structures are simplified to the two-body system connected to each other by

flexible wires. The free-body diagram of the two-body system is shown in Fig. 4.

From the dynamic equation of rotational motion,

$$J_1\ddot{\theta}_1 = 4rF\sin\phi + T_w$$
$$J_2\ddot{\theta}_2 = -4rF\sin\phi$$
$$J_w\dot{q}_w = T_w$$

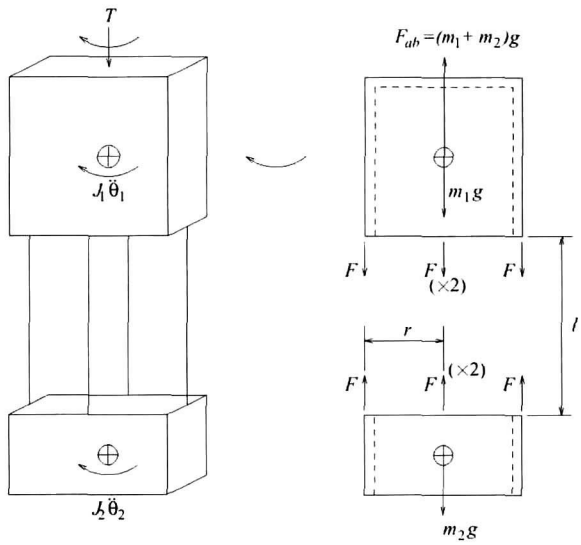



Fig. 4. Simplified free-body diagram

Table 2. Mechanical Properties of the System

| $J_1(kgm^2)$ | $J_2(kgm^2)$ | $J_w(kgm^2)$ | $m_2(kg)$ | $r(m)$ | $l(m)$ |
|---|---|---|---|---|---|
| 0.232 | 0.0783 | 2.387e-4 | 3.98 | 0.110 | 0.775 |

Table 3. Parameter Identification

| | Estimated | Identified |
|---|---|---|
| Zero( $z$) | $\pm 2.77i$ | $-0.0695 \pm 2.796i$ |
| Pole( $p$) | $0, \pm 3.21i$ | $0, -0.0145 \pm 3.157i$ |
| Torsional Stiffness( $K$) | 0.609 | 0.577 |

The subscript 1, 2 and $w$ represents the upper body, the lower body, and the reaction wheel, respectively. $l$ is the length of the wire, and $r$ represents the horizontal distance between the center of rotation and the wire link. $\phi$ represents the torsional deflection angle of the wire with respect to the vertical center plane, which can be restated with $\theta_1$ and $\theta_2$ for small $\phi$ and $(\theta_2-\theta_1)$.

$$l\phi = r(\theta_2-\theta_1)$$

$F$, which is the tensile force exerts on each wire, can be represented by the appendage mass $m_2$.

$$F=\frac{m_2 g}{4}$$

Defining a new constant K, which stands for the torsional stiffness,

$$K=\frac{m_2 g r^2}{l}$$

The system dynamic equations are completed as follows.

$$\begin{bmatrix}\dot{q}_1\\ \dot{q}_2\\ \dot{\theta}_1\\ \dot{\theta}_2\end{bmatrix}=\begin{bmatrix}0 & 0 & -K/J_1 & K/J_1\\ 0 & 0 & K/J_2 & -K/J_2\\ 1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\end{bmatrix}\begin{bmatrix}q_1\\ q_2\\ \theta_1\\ \theta_2\end{bmatrix}+\begin{bmatrix}1/J_1\\ 0\\ 0\\ 0\end{bmatrix}T_w$$

Poles and zeros of the natural oscillatory motion are located at,

$$z=\pm i\sqrt{\frac{m_2 g r^2}{J_2 l}}$$

$$p=0,\ \pm i\sqrt{\frac{(J_1+J_2)m_2 g r^2}{J_1 J_2 l}}$$

The numerical data, and identified parameters, which are used for simulation are provided in Table 2 and Table 3.

**Table 4. Nonlinear modeling of the reaction wheel**

| | |
|---|---|
| $T_w=-0.0075Nm$ | $(T_{cmd}<-0.0075Nm)$ |
| $T_w=T_{cmd}$ | $(-0.0075Nm\le T_{cmd}\le 0.0075Nm)$ |
| $T_w=0.0075Nm$ | $(0.0075Nm<T_{cmd})$ |

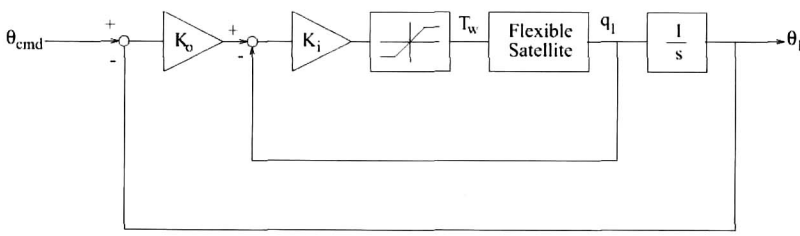



**Fig. 5. Single axis attitude dynamics of FSS**

## Controller Structure

The reliability of the systems is one of the most important factors for the space projects. Overall, the simpler parts offer the more reliability[15]. Therefore, classical PID control methods, which are very simple, are frequently applied for spacecraft computers of limited memory and speed.

The controller structure applied in this paper is a PD controller, which employs rate feedback and rotation angle feedback. Then the attitude controller acquires the function of flexibility control just by gain parameter variation, because the PD style is a very popular controller structure for attitude control of rigid spacecrafts. Hence, the reliability is hardly degraded. The confidential flight history of the classical controllers also encourages us to employ them.

With a simplified actuator/sensor modeling, the single axis attitude dynamics including a PD controller is given in Fig. 5. The root locus/classical design techniques usually do not provide full analyses for practical designs, while constrained optimization techniques such as co-evolutionary algorithms can handle various constraints and can consider nonlinearities on models[8]. In the next section, the gain parameters are selected by the accelerated co-evolutionary algorithm.

## Numerical Results

To apply optimization techniques to the design problem, the cost function is defined as the following integral form. The cost is zero when the response is identical to the reference model response, and increases as it deviates more from the reference.

$$J = 10 \int_{t_i}^{t_f} |\theta_1 - \theta_{ref}| \, dt$$

The second order system below is a good candidate reference model.

$$\frac{\theta_{ref}}{\theta_{cmd}} = \frac{\Omega_n^2}{s^2 + 2Z\Omega_n + \Omega_n^2}$$

The reference model with the natural frequency $\Omega_n = 0.3(rad/s)$ and the damping ratio $Z = 1$ provides a proper long-period dynamics. The reference system is chosen to be sufficiently relaxed that it wouldn't interfere the unmodeled wheel dynamics.

For vibration mode handling, the following constraint is considerd.

$$\zeta \geq 0.08$$

where $\zeta$ is the damping ratio of the vibration mode. As the identified natural frequency is $3.16(rad/s)$, the constraint lets the excited vibration be suppressed before the settling time.

The neural network structure for the CEALM is shown in Fig. 6. A simple type of feed-forward neural networks with a single hidden layer is used for function approximation. The cost and constraint violation information is stored in the networks. The hidden layer consists of twenty neurons, which have the hyperbolic tangent function as an activation function. The Levenberg-Marquardt algorithm is used for fast network training. The evolution parameters appear in Table 5.

Comparison of the cost histories shown in Fig.7 are based on 10 runs, respectively. In addition, the results from original co-evolutionary algorithms with similar size of offspring population are presented for a comparison. To analyze the convergence speed, an alternative expression of the cost, $J_E$ is defined as the following way.

$$J_E = |J - J_0|$$

where $J_0$ is the optimal cost. Then, as a reasonable convergence index, $g_C$, and $t_C$ is introduced, and examined. $g_C$ is the generation number for $J_E$'s convergence with a tolerance of $10^{-6}$, and $t_C$, the computation time to converge in the same tolerance.

**Table 5. ACEA Parameters for the FSS Problem**

| | |
|---|---|
| Number of Parents( $N_p$ ) | 20 |
| Number of Offspring( $N_o$ ) | 50 |
| Number of Extra Offspring( $N_{eo}$ ) | 50 |
| Selection Type | ( $\mu$ , $\lambda$ ) |
| Network Type | Single hidden layer/ Sigmoid |
| Number of Hidden Nodes | 20 |

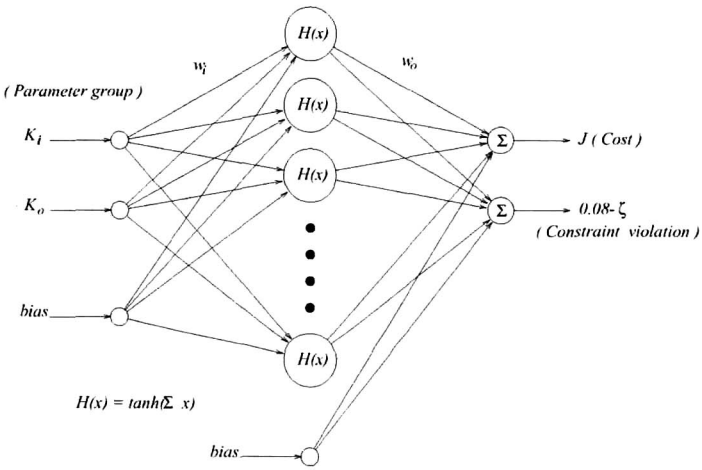
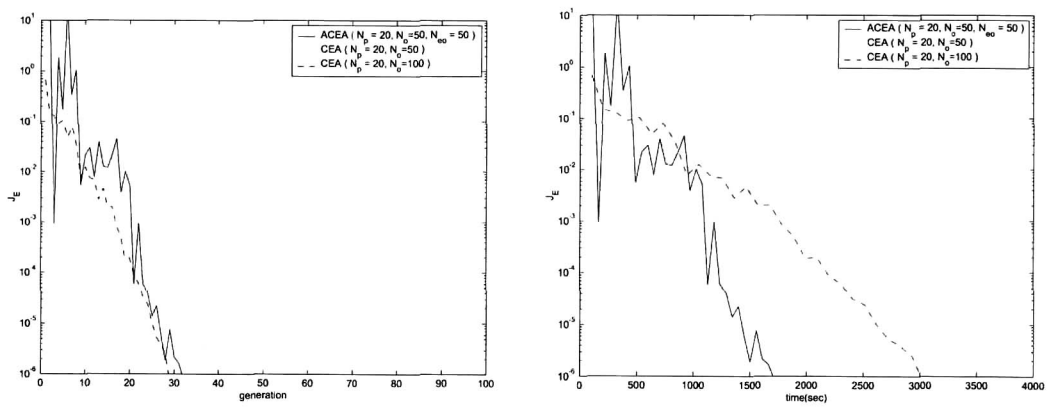



**Fig. 6. Neural network structures for Acc. CEALM**

**Table 6. Convergence Characteristics**

| | ACEA | CEA ($N_o$=50) | CEA ($N_o$=100) |
|---|---|---|---|
| $K_i$ | **1.0854** | 1.0854 | 1.085 |
| $K_o$ | **0.23107** | 0.23107 | 0.23107 |
| $J$(*Cost*) | **7.283779** | 7.283779 | 7.283779 |
| $g_C$(*gen*) | **32** | 29 | 53 |
| $t_C$(*sec*) | 1700 | 2800 | 3000 |
| *Constraint* | **Satisfied** | Satisfied | Satisfied |




**(a) Generation plot** **(b) Time plot**

**Fig. 7. $J_E$ convergence histories**

The parameters correctly converged to the estimated optimum in every trial. Compared to the original co-evolution history, the accelerated co-evolution history fluctuates more in earlier generations, for the newly proposed algorithm includes the neural network elements that can cause some approximation error. However, with the evolutionary selection processes, the fluctuation diminishes quickly.

Here, a new plot with another scale is presented for description of acceleration properties (Fig. 7(b)). As the computation time required for one generation differs from that of another, the

abscissa needs to be converted to the time scale. The scope of interest is highlighted on the cost history only. The computational results are obtained with the MATLAB on a Pentium III-500 desktop. The term *time* means the CPU time purely consumed by the computation process.

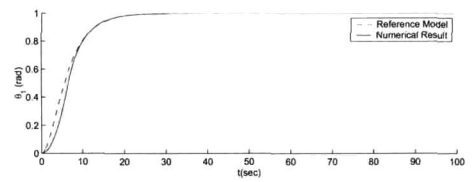
The generation plot implies that the newly proposed algorithm of ( $N_p$:$N_o$:$N_{eo}$ = 20:50:50), approximates the original algorithm of ( $N_p$:$N_o$ = 20 :100), with reduced computation time. The time plot tells that the new algorithm is the fastest among the three.

The results from the three are summarized in Table 6. Every figure in the table represents the median value.

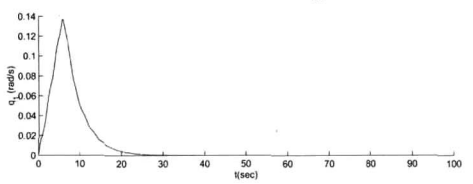
Now, the attitude controller is ready to be complete. Fig. 8 presents the simulation results for a step command, for . The optimal controller provides fine performance with excited vibration damped within 20 seconds. The damping ratio is slightly above 0.08, the constraint bound. The optimal controller designed by the newly proposed algorithm will be experimentally verified in the next section.

## Experimental Verification

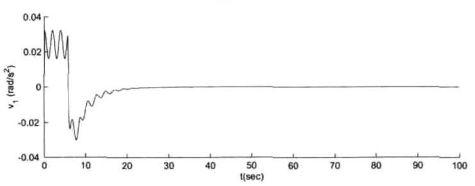
In the previous section, it is certified that the optimal controller provides so fine performance in the nominal condition. The cost is determined to have enough agility for usual missions while not interfering the unmodeled wheel dynamics. Beside the unmodeled wheel



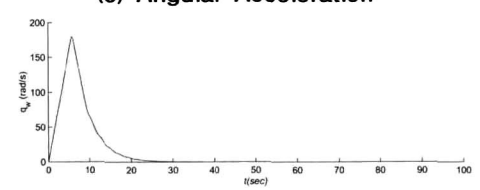

(a) Rotation angle




(b) Angular rate

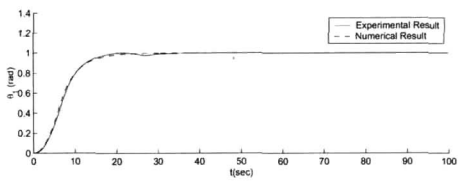



(c) Angular Acceleration

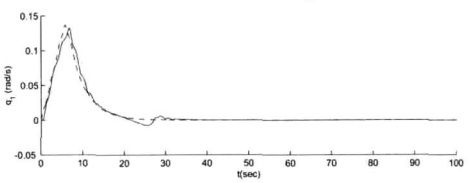



(d) Reaction wheel speed

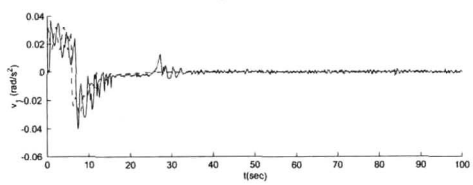
Fig. 8. Numerical results



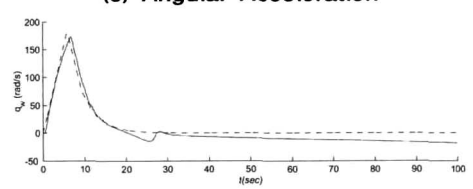

(a) Rotation angle




(b) Angular rate




(c) Angular Acceleration




(d) Reaction wheel speed

Fig. 9. Experimental results

dynamics, tiny friction in the air bearing system also generates small disturbance torque. The air bearing system is meant to provide a torque-free condition for tests, but it definitely cannot produce the perfect condition. The friction torque is not so serious that the imperfection can be regarded as a disturbance torque condition in the orbital environments. The modeling feasibility and robustness to the disturbance is confirmed in this section.

The experimental result for the optimal controller is presented in Fig. 9. The inquietude in the start quickly diminishes, and the vibration excited during the slew maneuver is also suppressed well. A small peak in the angular acceleration history(at about 25sec) is supposed to be the effect of the hidden wheel dynamics, and the wheel speed gradually decreasing after the settlement is for the friction in the air bearing system.

The experimental maneuver confidently follows the reference model, and the vibration absorption is well beyond the constraint bound. The gain parameters selected by the proposed algorithm work well for the large slew maneuver while rapidly suppressing the excited vibration.

## Concluding Remarks

In this paper, the accelerated co-evolutionary algorithms are proposed, and an attitude controller for a flexible satellite is designed by the proposed algorithm.

With a proper cost and constraint definition, the co-evolutionary algorithm provides confidential controller parameters. The accelerated algorithm lets the optimizing parameters converge faster than the original co-evolutionary algorithm does. The algorithm employs neural networks, which stores the cost and constraint information in its weights. The networks is trained whenever the cost is evaluated, and after sufficient training, the extra offspring group, whose cost is obtained by the neural network computation, is generated. As the neural network computation and training is composed of simple matrix calculations, increasing the offspring size by that way doesn't overload the computational load. Therefore, increased population size quickly converges the parameters without heavy computational load. The proposed algorithm fairly accelerated the convergence. Careful selection of neural network parameters or training algorithm will provide more rapid convergence characteristics.

As an application, an attitude controller for flexible space structures is designed by the proposed algorithm. The flexible structures are mathematically modeled ,and the design problem is formulated as a constrained optimization problem. The feasibility of design is numerically verified by the nonlinear simulation and experimentally demonstrated on a torque-free air bearing facility.

## References

1. G.D.Martin, and A.E.Bryson Jr., "Attitude Control of a Flexible Spacecraft," *Journal of Guidance, Control, and Dynamics,* Vol.1, No.1, pp.37-41, 1978.

2. B.Wie, and A.E.Bryson Jr., "Pole-Zero Modeling of Flexible Space Structures," *Journal of Guidance, Control, and Dynamics,* Vol.11, No.6, pp.554-561, 1988.

3. L.Meirovitch, H.Baruh, and H. Öz, "A Comparison of Control Techniques for Large Flexible Systems," *Journal of Guidance, Control, and Dynamics,* Vol.6, No.4, pp.302-310, 1983.

4. J.L.Junkins, Z.H.Rahman, and H.Bang, "Near-Minimum-Time Control of Distributed Parameter Systems: Analytical and Experimental Results," *Journal of Guidance, Control, and Dynamics,* Vol.14, No.2, pp.406-415, 1991.

5. J.Suk, J.Y.Moon, and Y.Kim, "Torque-Shaping Using Trigonometric Series Expansion for Slewing of Flexible Structures," *Journal of Guidance, Control, and Dynamics,* Vol.21, No.5, pp.698-703, 1998.

6. C.S.Park and M.J.Tahk, "A Co-evolutionary Minimax Solver and its Application to

Autopilot Design," in *Proc. AIAA Guidance, Navigation, and Control Conference,* Boston, USA, pp.408-415, Aug. 1998.

7. M.J.Tahk and B.C.Sun, "Coevolutionary Augmented Lagrangian Methods for Constrained Optimization," *IEEE Transactions on Evolutionary Computation,* Vol.4, No.2, pp.114-124, 2000.

8. J.Hur, and M.J.Tahk, "Robust Flight Control Design Using Eo-evolution With Considerations of Design Constraints," *3rd Asian Control Conference,* Shanghai, China, July 2000.

9. T.Bäck, Evolutionary Algorithms in Theory and Practice, Oxford University Press, 1996.

10. S.Haykin, *Neural Networks - A Comprehensive Foundation,* 2nd ed., Prentice Hall, 1999

11. C.S.Park, J.Hur, and M.J.Tahk, "Aircraft Control System Design Using Evolutionary Algorithm," *Jornal of the Korean Society for Aeronautical and Space Sciences,* Vol.27, No.4, pp.104-114, June 1999.

12. J.H.Kim, C.S.Park, and M.J.Tahk, "An Accelerated Co-evolutionary Algorithm Using Neural Networks," in *Proc. JSASS 37th Aircraft Symposium,* Tokyo, Japan, pp.717-720, Oct. 1999.

13. J.L.Junkins, and Y.Kim, *Introduction to Dynamics and Control of Flexible Structures,* AIAA, Washington, DC, 1993.

14. H.J.Buchanan, R.W.Schock, and H.B.Waites, "An On-orbit Experiment for Dynamics and Control of Large Structures," *Journal of Guidance, Control, and Dynamics,* Vol.7, No.5, pp.554-562, 1984.

15. W.J.Larson, and J.R.Wertz, *Space Mission Analysis and Design,* 2nd ed., Kluwer Academic Publishers, 1992.