Control-based CAS Design of QTW-UAVs
Using Particle Swarm Optimization*

Chiramathe NAMT, Koichi OKA, Akinori HARADA, and Masayuki SATO

1Department of Intelligent Mechanics and Aerospace Control, Kochi University of Technology, Kochi 782–0003, Japan
2Japan Aerospace Exploration Agency, Mitaka, Tokyo 181–0015, Japan

Key Words: Control Augmentation System, Structured $H_\infty$ Control, Quad-tilt-wing, Particle Swarm Optimization

Nomenclature

- $\phi$: roll angle
- $\tau_m$: tilt angle
- $m$: number of particles in PSO algorithm
- $n$: number of design variables
- $x_k^i$, $v_k^i$: position/velocity of $i$-th particle at iteration step $k$
- $C_0$: particle swarm inertia factor
- $C_1$: particle swarm cognitive factor
- $C_2$: particle swarm social scaling factor
- $r_k^i$, $\hat{r}_k^i$: uniform random numbers in $[0,1]$ at iteration step $k$ of $i$-th particle
- $x_k^{best,k}$: best solution of $i$-th particle up to iteration step $k$
- $x_k^{swarm}$: best solution in the entire swarm at iteration step $k$
- $x^*$: optimal solution
- $k$: iteration step number
- $k_{max}$: maximum iteration number
- $W(s)$: weighting function for sensitivity function
- $S(s)$: sensitivity function from attitude command to attitude error
- $K_{lat}$: admissible CAS gain set for lateral-directional motions
- $k_p$, $k_i$: proportional/integral gain
- $\omega_n$: cut-off frequency
- $DC$: direct current gain
- $HF$: high frequency gain

1. Introduction

Unmanned aerial vehicles (UAVs) have been widely researched in the last two decades. In particular, quad-tilt-wing UAVs (QTW-UAVs) have gained more attention as a possible tool for various applications such as surveillance, scientific measurement, etc. The Japan Aerospace Exploration Agency (JAXA) has therefore developed a series of QTW-UAVs (McART2, AKITSU3) and McART34). The flight test of AKITSU was conducted to verify the feasibility of QTW-UAVs in terms of practical size. It successfully flew, transforming from helicopter mode to airplane mode and vice versa; however, oscillatory motions were found in both longitudinal and lateral-directional motions. Similar motions were also found in the flight test of McART3. In particular, the oscillatory motion in the lateral-directional motion was sometimes large, and should therefore be suppressed for flight safety. This paper aims to be the first step to solve this problem. To this end, control augmentation system (CAS) gains were designed only for nominal models of the McART3 with SAS gains designed by Sato and Muraoka.6)

The drawbacks of the CAS design reported by Sato and Muraoka6) are twofold: time domain design specifications and large computational complexity. The details are given below. Controller gains are designed in a time domain for good tracking performance by minimizing the error between attitude (i.e., roll and pitch angles) and its step-type input command. The obtained gains worked well in flight tests, as reported by Sato and Muraoka. However, it is difficult to eliminate oscillatory motion completely, as shown in the full conventional flights of both QTW-UAVs. This is because it is not easy to impose time-domain constraints to suppress oscillatory motion. Next, CAS gains were optimized using a brute-force method (i.e., grid search optimization). When the ranges of controller gains are wide and/or the number of gains increases, large computational time is required. Currently, CAS is composed of only two control gains (i.e., proportional and integral) in the longitudinal and lateral-directional motions. However, there may be a problem if a complex CAS is adopted to enhance control performance.

To resolve the drawbacks mentioned above, we propose a design method with frequency domain constraints obtained by reducing computational complexity for optimization; that is, the CAS gains are designed by shaping the sensitivity function within the $H_\infty$ framework. Additionally, the Particle Swarm Optimization (PSO) method, which is one of the oriented search algorithms, is used as an alternative to the brute-force method to promote reduction of computational complexity. That is, the method of Maruta et al. is applied to fixed-structured $H_\infty$ controller design for the CAS of the QTW-UAV. As shown in Yavari et al., PSO is more effective than the Generic Algorithm (GA), which is one of the most famous meta-heuristic methods, in terms of compu-
and Eberhart,10) who were inspired by the social behaviors of birds flocking and fish schooling. The optimization process consists of three steps: initialization, solution update with very simple calculations, and termination judgement using pre-defined termination conditions. In this algorithm, candidates of the solution to the problem are represented by “particles.” Not only the local optimum obtained by each particle at all previous iteration steps, but also the global optimum obtained by whole particles (i.e., “swarm” at the current iteration step) are used to update the candidates.

The design procedures are roughly summarized below. In the initialization, a large number of particles are assigned randomly chosen solutions in a feasible search space, and the maximum iteration step number $k_{max}$ is defined. In the solution update, the best solution of $i$-th particles ($x_{i}^{\text{best},k}$) obtained previously and the best solution ($x_{\text{swarm}}^{\text{best}}$) among all particles at the current iteration step $k$ are calculated. Then, the velocity of each particle that denotes the direction and magnitude for updating the particle is calculated using these two kinds of positions ($x_{i}^{\text{best},k}$ and $x_{\text{swarm}}^{\text{best}}$) with random numbers ($r_{1}^{i}$, $r_{2}^{i}$). In the termination judgement, the decision for continuing the iteration is given.

Details with equations, which are cited from Maruta et al.,7) are given below. Suppose that the cost function is given as $f(x)$, then the problem is given as Eq. (2).

$$\min_{x \in \mathbb{R}^n} f(x),$$  

(2)

where the vector $x$ denotes a particle position, and the elements of which denote the design variables. The PSO algorithm uses a swarm of $m$ particles (i.e., $n$-dimensional vectors $x_{1}, x_{2}, \ldots, x_{m}$). The position of the $i$-th particle and its velocity are denoted, respectively, as $x_{i} := [x_{i,1}, x_{i,2}, \ldots, x_{i,n}]^T$ and $v_{i} := [v_{i,1}, v_{i,2}, \ldots, v_{i,n}]^T$ where $i \in \{1, 2, \ldots, m\}$. At iteration step $k$, $x_{i}^{\text{best},k}$ and $x_{\text{swarm}}^{\text{best}}$ are defined as follows:

$$x_{i}^{\text{best},k} := \arg \min_{x_{i}} f(x_{i}), \: 0 \leq j \leq k,$$

(3)

$$x_{\text{swarm}}^{\text{best}} := \arg \min_{x_{i}} f(x_{i}), \: 1 \leq i \leq n.$$  

(4)

Using these two solutions above, $x_{i}$ and $v_{i}$ are updated as follows:

$$x_{i}^{k+1} = x_{i}^{k} + v_{i}^{k+1},$$

(5)

$$v_{i}^{k+1} = C_{0} v_{i}^{k} + C_{1} r_{1}^{i}(x_{i}^{\text{best},k} - x_{i}^{k}) + C_{2} r_{2}^{i}(x_{\text{swarm}}^{\text{best}} - x_{i}^{k}).$$

(6)

$C_{0}$ values are the factors for inertia. Large values of $C_{1}$ will encourage the particles to move toward their local best solutions, while large values of $C_{2}$ will result in fast convergence to the global best solution.

The algorithmic steps of PSO are summarized below.

**Step 0:** Set $k_{max}$ and $k = 0$. Initialize $m$ particles with random positions in a feasible search space and evaluate the corresponding cost function values at each particle position. Calculate $x_{i}^{\text{best},0}$ and $x_{\text{swarm}}^{\text{best},0}$.

**Step 1:** If $k = k_{max}$, the algorithm stops with the solution

$$x_{i} := \arg \min_{x_{i}} f(x_{i}), \: i = 1, \ldots, m, \: j = 0, \ldots, k.$$  

Otherwise, go to Step 2.

**Step 2:** Apply Eqs. (5) and (6) to all particles and calculate the corresponding cost function values at each particle position. Update iteration number $k$, $k \rightarrow k + 1$. Calculate $x_{i}^{\text{best},k}$ and $x_{\text{swarm}}^{\text{best},k}$ before going back to Step 1.

Thanks to the attractive property that PSO does not require a smooth cost function with respect to design variables, nonlinear constraints (i.e., discontinuous cost functions) can be
PSO has been applied to various design problems such as non-convex optimization problems\(^{11}\), equality/inequality constraint optimization problems\(^{12}\), structured controller design problems\(^{7,13,14}\), etc.

3. Results

We first show the design results where the target sensitivity function was set the same as the one using CAS gains designed by Sato and Muraoka\(^6\) and then show the design results for oscillation suppression CAS. In both cases, \(W(s)\) was chosen as the low-pass filter using the “makeweight” function in the Matlab\(^8\) robust control toolbox\(^{15}\); the characteristics of which are parameterized with three parameters: \(DC\) gain, \(\omega_0\), and \(HF\) gain. We used 50 particles and set \(k_{\text{max}}\) as 50 in our design.

3.1. Basic design

To confirm the applicability of the proposed method, for the nominal McART3 models at all design tilt angles reported in Sato and Muraoka\(^6\), we tried to design CAS gains that have almost the same frequency characteristics as those of Sato and Muraoka\(^6\). Note that the controller gains reported by Sato and Muraoka\(^6\) were obtained by imposing time-domain constraints. Accordingly, obtaining almost the same frequency characteristics as Sato and Muraoka\(^6\) was not so straightforward. Appropriate \(W(s)\) are thus very important for the proposed method, and they were determined as follows: \(DC\) gains are first set as \(1.0 \times 10^{-6}\) to realize good tracking performance at low frequencies, as was done by Sato and Muraoka\(^6\). Next, the \(HF\) gains are set closely to the maximum peak gains of the closed-loop sensitivity function using the gains designed by Sato and Muraoka\(^6\). Finally, \(\omega_0\) is adjusted by trial-and-error to obtain the target tracking performance.

Following the abovementioned procedures, we obtained the CAS gains shown in Table 1(a) and (b). The optimized gains are almost the same as those reported by Sato and Muraoka\(^6\), except for the clean and 70 deg cases in the longitudinal motion and from 15–70 deg cases in the lateral-directional motions. However, it was confirmed that the step responses with those gains are almost the same as the gains reported by Sato and Muraoka\(^6\).

For reference, Fig. 2 shows the gain plots of the \(S(s)\) with the gains in Table 1, and the inverse of our \(W(s)\) for the longitudinal motion and lateral-directional motions at a wing tilt angle of 15 deg. These results confirm the applicability of our method.

3.2. Extension to oscillation suppression CAS

As an extension of the basic design, we address oscillation suppression CAS design for lateral-directional motion. We also addressed the same problem for longitudinal-direction motion; however, we couldn’t design suitable CAS gains due to the strictly defined admissible gain intervals for admissible CAS gains.

\(W(s)\) for oscillation suppression CAS gains are determined in almost the same manner as the basic design. However, \(HF\) gains are adjusted lower than the maximum peak gains of the \(S(s)\) by several trial-and-error using CAS gains designed by Sato and Muraoka\(^6\). Then, \(\omega_0\) was adjusted to enhance tracking performance as well.

Using the \(W(s)\) obtained as explained above, we designed CAS gains for lateral-directional motion. In particular, the

| \(\tau_w\) | \(K_{\text{in}}\) in Sato and Muraoka\(^6\) | Gains in Sato and Muraoka\(^6\) | PSO | \(\omega_0\) (rad/s) | \(HF\) |
|---|---|---|---|---|---|
| Gains in Sato and Muraoka\(^6\) | \(k_p, k_i\) | \(k_p', k_i'\) | \(kp/C_18\) | | |
| c | \([0, -70]\) | \([-70, -33]\) | \([-43, -2]\) | 0.9 | 1.1 |
| 0 | \([0, -100]\) | \([-100, -100]\) | \([-100, -98]\) | 0.95 | 2.6 |
| 15 | \([0, -150]\) | \([-150, -100]\) | \([-146, -92]\) | 0.3 | 1.9 |
| 30 | \([0, -150]\) | \([-150, -100]\) | \([-150, -100]\) | 0.3 | 1.9 |
| 50 | \([0, -100]\) | \([-70, -100]\) | \([-70, -67]\) | 0.4 | 1.75 |
| 70 | \([0, -50]\) | \([-50, -40]\) | \([-49, -20]\) | 0.97 | 5.5 |
| 90 | \([0, -50]\) | \([-40, -40]\) | \([-44, -40]\) | 1.5 | |

| \(\tau_w\) | \(K_{\text{in}}\) in Sato and Muraoka\(^6\) | Gains in Sato and Muraoka\(^6\) | PSO | \(\omega_0\) (rad/s) | \(HF\) |
|---|---|---|---|---|---|
| Gains in Sato and Muraoka\(^6\) | \(k_p, k_i\) | \(k_p', k_i'\) | \(ki/C_{18}\) | | |
| c | \([0, -100]\) | \([-100, -50]\) | \([-94, -50]\) | 2 | 2 |
| 0 | \([0, -100]\) | \([-100, -50]\) | \([-94, -48]\) | 0.65 | 3 |
| 15 | \([0, -100]\) | \([-100, -50]\) | \([-79, -50]\) | 0.6 | 2 |
| 30 | \([0, -120]\) | \([-88, -50]\) | \([-106, -48]\) | 2 | 3 |
| 50 | \([0, -80]\) | \([-80, -50]\) | \([-64, -50]\) | 1.7 | 1.6 |
| 70 | \([0, -50]\) | \([-50, -40]\) | \([-27, -39]\) | 0.6 | 1.5 |
| 90 | \([0, -50]\) | \([-50, -40]\) | \([-44, -40]\) | 0.3 | 1.5 |

As an extension of the basic design, we address oscillation suppression CAS design for lateral-directional motion. We also addressed the same problem for longitudinal-direction motion; however, we couldn’t design suitable CAS gains due to the strictly defined admissible gain intervals for admissible CAS gains.

\(W(s)\) for oscillation suppression CAS gains are determined in almost the same manner as the basic design. However, \(HF\) gains are adjusted lower than the maximum peak gains of the \(S(s)\) by several trial-and-error using CAS gains designed by Sato and Muraoka\(^6\). Then, \(\omega_0\) was adjusted to enhance tracking performance as well.

Using the \(W(s)\) obtained as explained above, we designed CAS gains for lateral-directional motion.
gain design in existing design methods for quad-tilt-wing unmanned aerial vehicles (QTW-UAVs). In contrast to the method previously used, design requirements are given in the frequency domain. To be more specific, weighted sensitivity function from attitude command to attitude error is required to satisfy the $H_{\infty}$ norm constraint. The applicability of the method proposed was first verified through CAS design for the McART3, the target sensitivity function of which was obtained using the gains reported in an existing paper. Then, as an extension, the authors addressed oscillation suppression CAS design. In exchange for slightly slower responses, oscillatory-suppression CAS gains were obtained in the latter problem.

4. Conclusion

In this paper, the authors have proposed a control augmentation system (CAS) design that utilizes Particle Swarm Optimization (PSO) algorithm to address the drawbacks of oscillatory motion and large numerical complexity in controller gain design. This method might be useful for designing controllers for quad-tilt-wing unmanned aerial vehicles and other systems with similar characteristics. The authors demonstrated the effectiveness of their method through simulations and experiments, showing improved performance compared to existing methods. Future work could involve further optimization and validation of the method in real-world applications.

References

1) Cox, T. H., Nagy, C. J., Skoog, M. A., and Somers, I. A.: Civil UAV Capability Assessment, NASA TR, Dec. 2004.
2) Hassanalian, M. and Abdelkeith, A.: Classifications, Applications, and Design Challenges of Drones: A Review, Prog. Aerospace Sci., 91 (2017), pp. 99–131.
3) Unmanned Aircraft Systems Roadmap. 2005–2030, U.S. Dept. of Defense, Office of the Secretary of Defense, 2005, https://www.fas.org/irp/program/collect/uav.roadmap2005.pdf (accessed October 20, 2013).
4) Muraoka, K., Okada, N., Kubo, D., and Sato, M.: Transition Flight of Quad Tilt Wing VTOL UAV, 28th International Congress of the Aeronautical Sciences, ICAS 2012-11.1.3, Bonn, Germany, 2012.
5) Sato, M. and Muraoka, K.: Flight Controller Design and Demonstration of Quad-Tilt-Wing Unmanned Aerial Vehicle, J. Guid. Control Dyn., 38 (2015), pp. 1071–1082.
6) Sato, M. and Muraoka, K.: Flight Controller Design for Small Quad Tilt Wing UAV, J. Jpn. Soc. Aeronaut. Space Sci., 64 (2016), pp. 79–82 (in Japanese).
7) Maruta, I., Kim, T., and Sugie, T.: Fixed-structure $H_{\infty}$ Controller Synthesis: A Meta-heuristic Approach using Simple Constrained Particle Swarm Optimization, Automatica, 45 (2009), pp. 553–559.
8) Yavari, S., Zoue, M. J. V., Mokhtarzadeh, M., and Mohammadmazdeh, A.: Comparison of Particle Swarm Optimization and Genetic Algorithm in Rational Function Model Optimization, Int. Arch. Photogrammetry Remote Sensing Spatial Information Sci., 39, B1 (2012), pp. 281–284.
9) Skogestad, S. and Postlethwaite, I.: Multivariable Feedback Control Analysis and Design, Second Edition, John Wiley & Sons Press, New York, 2001, pp. 55–57.
10) Kennedy, J. and Eberhart, R. C.: Particle Swarm Optimization, Proceeding of 1995 IEEE International Conference on Neural Networks, Perth, WA, Australia, 1995.
11) Banks, A., Vincent, J., and Anyakoha, C.: A Review of Particle Swarm Optimization. Part I: Background and Development, Natural Comput., 6 (2007), pp. 467–484.
12) Sedlaczek, K. and Eberhard, P.: Using Augmented Lagrangian Particle Swarm Optimization for Constrained Problems in Engineering, Structural and Multidisciplinary Optimization, 32 (2006), pp. 277–286.
13) Philippe, F.: Robust Control Optimization with Metaheuristics, John Wiley & Sons Press, USA, 2017, pp. 238–248.
14) Tuvayanond, W. and Panichkun, M.: Position Control of a Pneumatic Surgical Robot using PSO Based 2-DOF $H_{\infty}$ Loop Shaping Structured Controller, Mechatronics, 43 (2017), pp. 40–55.
15) MATLAB and Robust Control Toolbox Release 2017b, The MathWorks, Inc., Natick, United States, 2017.