Adaptive Particle Swarm Optimization Applied to Aircraft Control

Ouboti Djaneye-Boundjou
Advisor: Raúl Ordoñez; Other Participant: Lance Jacobsen
Department of Electrical and Computer Engineering, University of Dayton

Pushing the Boundaries of Flight

• Can you imagine being able to travel overseas in less than an hour?
• Air-breathing Hypersonic Vehicles (AHV’s) could make it possible.

Feedback Alignment

Fig. 1: US Air Force’s X-51A Waverider. (a) Angle of attack and (b) AoA in a plane

Synopsys of Present Study

• Consider longitudinal dynamics of a fixed wing aircraft with rigid frame.
• Simulations are run at much lower speed.
• Designed controllers:
  – Proportional-Integral (PI) controller to control forward velocity $V_f$.
  – Gain-scheduled Proportional-Integral-Differential (PID) like controller to control flight path angle $\gamma$.
• Automated tuning process via optimization.

Fixed Wing Aircraft with Rigid Frame

• Longitudinal dynamics [7]:
  \[
  \frac{dV_f}{dt} = \frac{1}{M} \left( C_{D,0}(\alpha) - \frac{D}{m} \right) + g \sin(\theta - \alpha),
  \]
  \[
  \alpha = \frac{1}{M} \left( -T \sin\alpha + L_z + mg\cos(\theta - \alpha) \right) + \psi,
  \]
  \[
  \frac{\theta}{\phi} = \frac{\dot{\psi}}{\dot{\phi}}.
  \]
• Aerodynamic forces:
  \[
  D = \frac{1}{2} \rho V_f^2 S_C D (\alpha, h, q),
  \]
  \[
  L_z = \frac{1}{2} \rho V_f^2 S_C L_z (\alpha, h, q),
  \]
  \[
  M = \frac{1}{2} \rho V_f^2 S_C M (\alpha, h, q).
  \]
• States: $x_t = [V_f, \alpha, \beta, \gamma, \delta]$. $\dot{x}_t$=\( [V_f, \alpha, \beta, \gamma, \delta] \)
• Nonlinear Multiple Input Multiple Output (MIMO) system:
  – 2 control inputs: thrust $T$ and elevator deflection $\delta_E$.
  – 2 outputs: forward speed $V_f$ and flight path angle $\gamma = \theta - \alpha$.

Aircraft Control

• Complex system to control: $C_{D,0}$, $C_{L,0}$, and $C_{D,\gamma}$ bear convoluted couplings between system’s states and control inputs.
• Literature: linear and nonlinear controllers [5, 9, 11] under various conditions and assumptions.
• PI and PID controllers are used here: structurally simple and easily implementable.

Control Design

• Drive $x_t$ at $t = [V_f, \alpha, \beta, \gamma]$ to desired $[\dot{x}_t] = [\ddot{V}_f, \dot{\alpha}, \dot{\beta}, \dot{\gamma}]^\top$.
• Error $e = [\ddot{V}_f, \dot{\alpha}, \dot{\beta}, \dot{\gamma}] = \gamma - \gamma_{des} = \gamma - \gamma_{ref}$.
• Thrust $T$: PI controller to control $V_f$.
  \[
  T(t) = K_{Pf} \gamma_{ref} + K_{I} \int_{0}^{t} \gamma_{ref}(s) ds.
  \]
• Elevator deflection $\delta_E$: gain-scheduled PID-like controller, with $V_f$ being the scheduling variable, to control $\gamma$.
  \[
  \delta_E(t) = K_{P\gamma} \gamma_{ref} + K_{I\gamma} \int_{0}^{t} \gamma_{ref}(s) ds + K_{D\gamma} \dot{\gamma}_{ref}(t). \]
• Input bounds (saturation if needed): $T \geq 0$ and $-20^\circ \leq \delta_E \leq 20^\circ$.

Tuning

• Simulation-based tuning:
  – Design velocities $V_{f,\text{ref}} = [V_f, V_f, \ldots, V_f]$. (Table 1: Simulation-based tuning PID gains)
  – For $t = 31$, $K_{P\gamma} = K_{I\gamma} = -15$.

Preliminaries

• Training: desired profile to track for each $V_f$ in $V_{f,\text{ref}}$.
  \[
  \begin{align*}
  V_{f_{\text{opt}}} &= [V_f, \alpha, \beta, \gamma] \quad \text{and}, \quad \gamma_{\text{opt}} = [0^\circ, 1^\circ, 1^\circ, 1^\circ] \quad \text{for} \ t_{\text{ref}} = [V_f, t_1].
  \end{align*}
  \]
• Condition 1: keep within $\theta_{\text{ref}} \leq \theta$ and $\gamma \leq \alpha$.
• Condition 2: for $t_{\text{ref}} \leq t \leq t_{\text{ref}}$, $\gamma_{\text{ref}}$ is ultimately bounded.
• $K_\gamma$ is an alternative solution (AS) if Conditions 1 and 2 are satisfied.
• Store best solution for each $V_f$ in $K_{\gamma,\text{opt}} \in \mathbb{R}^{1\times m}$, $m = 1, 2, \ldots, 4$.

Performance Measures

• $\dot{c}_{\text{opt}}, \dot{c}_{\text{ref}}$, number of samples, $6$ its mean, $9$ its Fast Fourier Transform (FFT), $5$ the FFT of $\nu(t)$.
• $\nu(t)$ is a FF of $\nu(t)$, $\nu(t)$ denote frequency, and $\nu(t)$ is a cubic spline polynomial.
• Time-domain criteria: INTSE (IAPSE), for different $c_{\text{opt}}$ and $c_{\text{ref}}$ given by $J = \int_{t_1}^{t_2} \left| \dot{c}_{\text{opt}}(t) - \dot{c}_{\text{ref}}(t) \right|^2 dt$.
• MRC given by $J = m \max \left\{ \left| \dot{c}_{\text{opt}}(t) - \dot{c}_{\text{ref}}(t) \right| : t \in [t_1, t_2] \right\}$.
• SE: $10, 8, 2$ given by $J = \frac{1}{2} \left| \dot{c}_{\text{opt}}(t) - \dot{c}_{\text{ref}}(t) \right|^2$.
• 81 total criteria $C_1 \geq 0$ for each AS $K_\gamma$.

Stable Adaptive PSO (APSO)

• PSO: Population-based (N-dimensional) stochastic search inspired by biological systems (bird flocking, fish schooling).
• Apply PSO-based tuning to full dynamics with flexible states.

Bookkeeping

• When training, for each $V_f$:
  – Create and update memory bank $U_{\text{opt}}$, containing all AS’s, their $81$ $C_1$ and PSO parameter indices.
  – Find maximum performance measure $M_{\text{opt}}^f$ across each $f$ criterion at each PSO iteration $f$.

Designed Cost Function

• Normalization: $d = \min \{ N F_{\text{opt}} = 1, M_{\text{opt}} = 0, M_{\text{opt}} = 0 \}$.
• $C_{\text{opt}}$ to minimize: $f\left(K_\gamma\right) = \sum_{t=1}^{\infty} \frac{\nu(t)}{\nu(t)^2}$.
• Decision Algorithm:
  – When a new $M_{\text{opt}}^f$ is found, use data in $U_{\text{opt}}$ to recompute the cost of past and present AS’s and recompute the ASPO accordingly.

POSO-based PID Tuning

• Initial conditions: $\gamma_{\text{init}} = [V_f, 0, 0, 0, 0, 0, 0, 0, 1000]$.
• Training: $N = 10$, $t_1 = 150$, $V_f_{\text{ref}} = 50$, $t_2 = 10$, $T = 30$, $T = 30$, $T = 30$, $V_f_{\text{ref}} = 20$, $\dot{\gamma} = 16.5, \alpha = d = 4, 22$, $\dot{\theta} = 1$. For $d = 3, 5, 27, 30, 37, 38, 44, 46, 47, 48, 49, 50, 51, 52, 54, 55, 56, 57, 58, 60, 61, 62, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81$, and $\dot{\theta}_{\text{ref}} = 0$.
• Testing: $V_f_{\text{ref}} = 30$.

Training Results

• Testing: $V_f_{\text{ref}} = 30$.

Future Work

• Thorough testing.
• For possible better performance: find appropriate design parameter combination, investigate Multi-Objective Optimization.
• Apply PSO-based tuning to full dynamics with flexible states.
• Develop robust controller since PID control lacks robustness.

References