11th ANKARA INTERNATIONAL AEROSPACE CONFERENCE **AIAC-2021-159**
8-10 September 2021 - METU, Ankara TURKEY

APPLICATION OF CONSTANT ADAPTIVE NEWTON METHOD TO AIRCRAFT TRIM PROBLEM

Salih Volkan Özkan* and Ozan Tekinalp †
Middle East Technical University
Ankara, Turkey

ABSTRACT

Simulation models containing aerodynamic database, propulsion system, sensor and actuator dynamics of a fighter is considered. Trim conditions for a specific flight envelope are obtained using classical Newton method as well as Adaptive Newton method. The success of these algorithms in finding trim flight conditions are evaluated and presented.

NOMENCLATURE

$\delta_{ht}, \delta_{aileron}, \delta_{rudder}, \delta_{throttle}$ = Horizontal tail, aileron, rudder, and throttle delta deflections

α = Angle of attack

β = Angle of sideslip

N_z = Normal acceleration in body z direction

$Mach$ = Mach Number

p, q, r = Body axis x,y,z angular rates

u, v, w = Body axis x,y,z velocities

m = Mass of the aircraft

ϕ, θ, ψ = Roll, pitch, and yaw angles (Euler Angles)

R, M, N = Body axis x,y,z moments acting on the center of gravity

X, Y, Z = Body axis x,y,z forces acting on the center of gravity

I = Aircraft inertia matrix

$(.)_E$ = In earth axis

*Ms Student in Aerospace Engineering Department, Email: ozkan.volkan@metu.edu.tr
†Prof. in Aerospace Engineering Department, Email: tekinalp@metu.edu.tr

$(.)_I$ = In inertial axis

$(.)_B$ = In body axis

INTRODUCTION

Aircraft equations of motion are nonlinear. The nonlinearities come from kinematics, as well as from the aerodynamic, and propulsive forces. These forces change with for example, Mach number, angle of attack, control surface deflections, etc. The usual method for the evaluation of stability of a nonlinear system is to linearize these equations around an equilibrium, called trim point. In aerospace literature, trim points are the steady flight conditions. Since number of trim equations are less than unknowns, some of the physically meaningful unknowns must be specified. Then, these nonlinear algebraic equations must be solved.

Linearization of nonlinear flight equations are obtained by evaluating the Jacobian matrices around the trim condition. Let $x \in \mathbb{R}^m$, $u \in \mathbb{R}^n$, and $y \in \mathbb{R}^p$ then the linearized equations are:

$$\dot{x} = Ax + Bu \tag{1}$$

$$y = Cx + Du \tag{2}$$

where,

$$A = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_m} \end{bmatrix} B = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \cdots & \frac{\partial f_1}{\partial u_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial u_1} & \cdots & \frac{\partial f_m}{\partial u_n} \end{bmatrix} C = \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \cdots & \frac{\partial g_1}{\partial x_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial g_p}{\partial x_1} & \cdots & \frac{\partial g_p}{\partial x_m} \end{bmatrix} D = \begin{bmatrix} \frac{\partial g_1}{\partial u_1} & \cdots & \frac{\partial g_1}{\partial u_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial g_p}{\partial u_1} & \cdots & \frac{\partial g_p}{\partial u_n} \end{bmatrix} \tag{3}$$

Elements of system matrices A, B, C, and D are obtained via famous small perturbation theory around the trim condition and these are denoted as,

$$A_{i,j} = \frac{f(x+\Delta x, u) - f(x-\Delta x, u)}{2\Delta x};\ i = 1, \cdots, m;\ j = 1, \cdots, m \tag{4}$$

$$B_{i,j} = \frac{f(x, u+\Delta u) - f(x, u-\Delta u)}{2\Delta u};\ i = 1, \cdots, m;\ j = 1, \cdots, n \tag{5}$$

$$C_{i,j} = \frac{g(x+\Delta x, u) - g(x-\Delta x, u)}{2\Delta x};\ i = 1, \cdots, p;\ j = 1, \cdots, m \tag{6}$$

$$D_{i,j} = \frac{g(x, u+\Delta u) - g(x, u-\Delta u)}{2\Delta u};\ i = 1, \cdots, p;\ j = 1, \cdots, n \tag{7}$$

In this study, utilization of Matlab environment in the calculation of the linearized equations at different trim flight conditions is addressed. For given trim conditions, linearized system matrices can be constructed in this fashion. MATLAB has a built-in function feval [Mathworks Inc , 2017] which can be used to calculate the Jacobian matrices with respect to states and inputs.

$$Derivatives = feval(model,\ t_0, x_0, u_0, 'derivs')$$
$$Outputs = feval(model,\ t_0, x_0, u_0, 'outputs')$$

where model is the Simulink model name, t_0 linearization time, x_0 initial state vector, u_0 initial input vector. After constructing these matrices, we can combine system of equations as follows,

$$\begin{bmatrix} \dot{x} \\ y \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix}, H = \begin{bmatrix} A & B \\ C & D \end{bmatrix}, z = \begin{bmatrix} x \\ u \end{bmatrix}, b = \begin{bmatrix} \dot{x} \\ y \end{bmatrix} \tag{8}$$

Finding the trim conditions require solving set of algebraic equations iteratively given in (8). There are m states, m state derivatives, n inputs and p outputs. From the equation (8), it is obvious that, the states and inputs are unknown and the outputs and state derivatives are known variables. These known variables are called freeze conditions while unkown variables are called float conditions. Therefore, in order to fully construct the problem, number of freeze conditions should be equal to the number of float conditions which means the matrix H is full rank and invertible. The objective function then can be defined as in (9).

$$\begin{aligned} \|P(z)\|_2 &= \|Hz - b\|_2 \\ &= 0 \end{aligned} \tag{9}$$

This type of linear system of equations can be solved by employing Newton's method.

Algorithm 1: Newton's Method

Result: z
k = 0;
Initialize z_0;
$p_k = \|P(z_k)\|_2$;
while $\|P(z_k)\|_2 > 10^{-8}$ **do**
  $x_k = \underset{P'(z_k)x=P(z_k)}{\operatorname{argmin}} \|x\|_2$;
  $\gamma_k = 1$;
  $p_{k+1} = \|P(z_k - \gamma_k x_k)\|_2$;
  $z_{k+1} = z_k - \gamma_k x_k$;
  $k + 1$;
end

F16 Aircraft Simulation Model

Newton's method is applied on 6-Degrees of Freedom (6-DoF) F-16 simulation model (Figure 1) which only includes aerodynamic coefficients [Nguyen et al. , 1979], atmosphere model, engine model [Huo , 2012], equations of motion, and simple actuator dynamics. This model does not include high number of nonlinearities, e.g., rate limiters, dead-bands, control system, sensor dynamics, piecewise linear functions, etc.

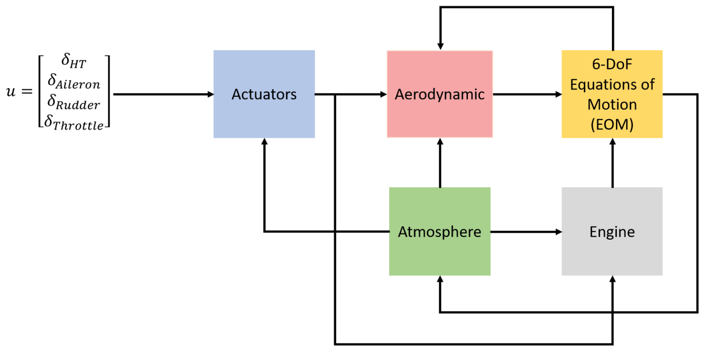


Figure 1: F16 6-DoF Simulation Model

Advanced Jet Trainer Aircraft Simulation Model

Newton's method is also applied on advanced jet trainer aircraft model (Figure 2) which includes flight control software, high fidelity aerodynamic, actuator, sensor, engine models with high number of nonlinear elements.

Equations of Motion

Equations of motion [Stevens et al. , 2016] includes translational and rotational motion of a rigid

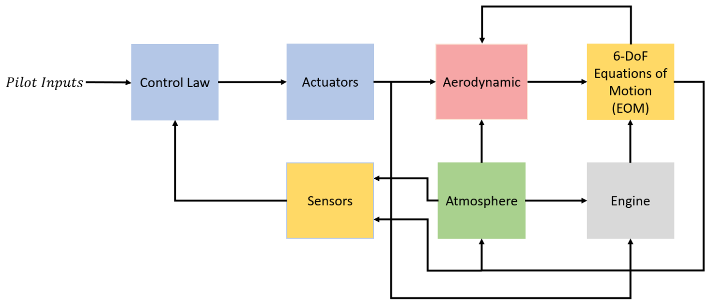


Figure 2: Advanced Jet Trainer 6-DoF Simulation Model

body, or Newton-Euler equations. Forces and moments acting on the body center of gravity are related to the motion of the body-axis are written. Body and inertial axes are defined in (Figure 3).

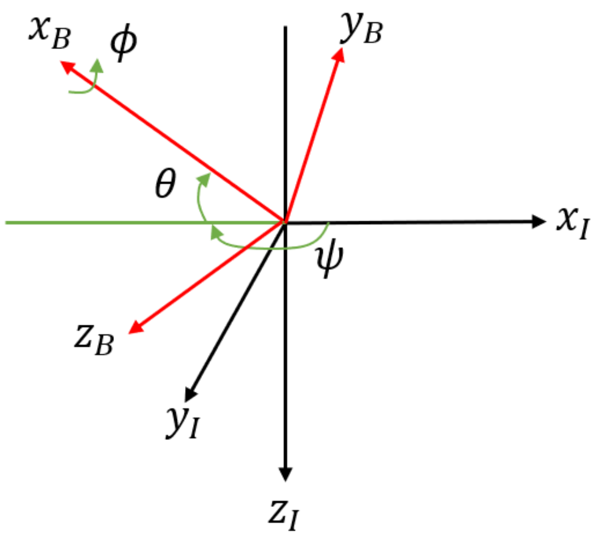


Figure 3: Inertial and Body Frames

Transformation from inertial to body axis frame via Z-Y-X Euler sequence are written as:

$$\begin{bmatrix} x_B \\ y_B \\ z_B \end{bmatrix} = R(\psi)R(\theta)R(\phi) \begin{bmatrix} x_I \\ y_I \\ z_I \end{bmatrix} \tag{10}$$

$$R(\psi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & \sin\psi \\ 0 & -\sin\psi & \cos\psi \end{bmatrix} R(\theta) = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} R(\phi) = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{11}$$

$$\begin{aligned} T_I^B &= R(\psi)R(\theta)R(\phi) \\ &= \begin{bmatrix} \cos\psi\cos\theta & \sin\psi\cos\theta & -\sin\theta \\ \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi & \sin\psi\sin\theta\sin\phi + \cos\phi\cos\psi & \cos\theta\sin\phi \\ \cos\psi\sin\theta\cos\phi + \sin\phi\sin\psi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi & \cos\theta\cos\phi \end{bmatrix} \end{aligned} \tag{12}$$

Transformation matrix is an orthogonal matrix which means its transpose is equal to its inverse.

$$(T_I^B)^T = T_B^I \tag{13}$$

Translational equations in body axis,

$$\begin{aligned}\dot{\vec{v}}_B &= T_I^B \dot{\vec{v}}_I - \vec{\omega}_B \times \vec{v}_B \\ &= \vec{F}_B - \vec{\omega}_B \times \vec{v}_B\end{aligned} \tag{14}$$

$$\begin{bmatrix}\dot{u}\\ \dot{v}\\ \dot{w}\end{bmatrix} = \frac{1}{m}\begin{bmatrix}X\\ Y\\ Z\end{bmatrix} - \begin{bmatrix}p\\ q\\ r\end{bmatrix} \times \begin{bmatrix}u\\ v\\ w\end{bmatrix} \tag{15}$$

where m is the mass of the aircraft, X, Y, Z are the forces acting on the center of gravity in the body axis frame. Navigational equations are defined in inertial axis and using the transformation matrix,

$$\begin{bmatrix}\dot{x}_E\\ \dot{y}_E\\ \dot{z}_E\end{bmatrix} = T_B^I \begin{bmatrix}u\\ v\\ w\end{bmatrix} \tag{16}$$

Rotational equations in body axis,

$$\begin{bmatrix}\dot{p}\\ \dot{q}\\ \dot{r}\end{bmatrix} = I^{-1}\left[\begin{bmatrix}R\\ M\\ N\end{bmatrix} - \begin{bmatrix}p\\ q\\ r\end{bmatrix} \times I \begin{bmatrix}p\\ q\\ r\end{bmatrix}\right] \tag{17}$$

where I is the inertia matrix of the aircraft, R, M, N are the moments acting on the center of gravity in the body axis frame. Kinematic equations are,

$$\dot{\phi} = p + \tan\theta(q\sin\theta + r\cos\phi) \tag{18}$$

$$\dot{\theta} = q\cos\phi + r\sin\phi \tag{19}$$

$$\dot{\psi} = \frac{q\sin\phi + r\cos\phi}{\cos\theta} \tag{20}$$

Trim Types

Since simulation model, equations of motion and linearized equations are shown, now we can define states, state derivatives, inputs and outputs which are going to be used in trim problem.

$$\begin{bmatrix}\dot{u}\\ \dot{v}\\ \dot{w}\\ \dot{x}_E\\ \dot{y}_E\\ \dot{z}_E\\ \dot{p}\\ \dot{q}\\ \dot{r}\\ \dot{\phi}\\ \dot{\theta}\end{bmatrix} = A\begin{bmatrix}u\\ v\\ w\\ x_E\\ y_E\\ z_E\\ p\\ q\\ r\\ \phi\\ \theta\end{bmatrix} + B\begin{bmatrix}\delta_{ht}\\ \delta_{aileron}\\ \delta_{rudder}\\ \delta_{throttle}\end{bmatrix} \tag{21}$$

$$\begin{bmatrix} Mach \\ N_z \end{bmatrix} = C \begin{bmatrix} u \\ v \\ w \\ x_E \\ y_E \\ z_E \\ p \\ q \\ r \\ \phi \\ \theta \end{bmatrix} + D \begin{bmatrix} \delta_{ht} \\ \delta_{aileron} \\ \delta_{rudder} \\ \delta_{throttle} \end{bmatrix} \tag{22}$$

Using the equations in (21) and (22), freeze and float conditions can be set for different trim types. In this paper, three different trim types will be evaluated which are, wings and level, coordinated turn, and pull up/push down.

Wings and Level Trim:

Wings and level trim corresponds to the cruise flight of the aircraft. This trim type requires freeze and float conditions given in Table 1 [Roskam , 2001].

Freeze Conditions		Float Conditions	
State Derivatives	Outputs	States	Inputs
$\dot{u}, \dot{v}, \dot{w}, \dot{y}_E, \dot{z}_E, \dot{p}, \dot{q}, \dot{r} = 0$	Mach	u, v, w, ϕ, θ	$\delta_{ht}, \delta_{aileron}, \delta_{rudder}, \delta_{throttle}$

Table 1: Straight, Symmetric, Wings Level Flight Trim

Coordinated Turn Trim:

Coordinated turn trim corresponds to aircraft in a turn at a constant altitude, velocity, and load factor/turn rate. This trim type requires freeze and float conditions given in Table 2 [Roskam , 2001].

Freeze Conditions		Float Conditions	
State Derivatives	Outputs	States	Inputs
$\dot{u}, \dot{v}, \dot{w}, \dot{z}_E, \dot{p}, \dot{q}, \dot{r}, \dot{\phi}, \dot{\theta} = 0$	Mach	u, v, w, q, r, θ	$\delta_{ht}, \delta_{aileron}, \delta_{rudder}, \delta_{throttle}$

Table 2: Coordinated Turn Trim

Pull Up/Push Down Trim:

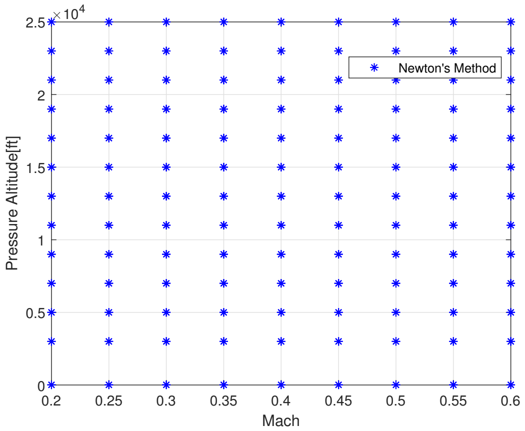
Pull up/push down trim corresponds to aircraft in a vertical turn at a constant load factor. This trim type requires freeze and float conditions given in Table 3 [Roskam , 2001].

Freeze Conditions		Float Conditions	
State Derivatives	Outputs	States	Inputs
$\dot{u}, \dot{v}, \dot{w}, \dot{p}, \dot{q}, \dot{r}, \dot{\phi} = 0$	Mach, N_z	u, v, w, q, θ	$\delta_{ht}, \delta_{aileron}, \delta_{rudder}, \delta_{throttle}$

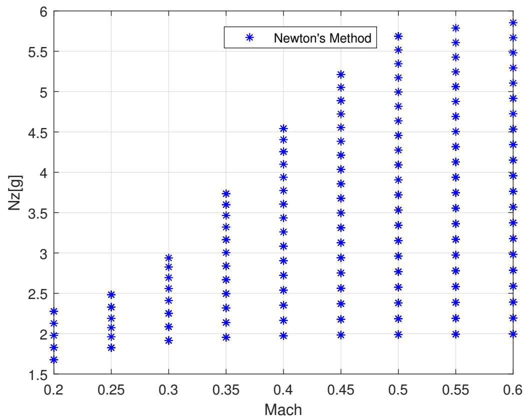
Table 3: Pull Up/Push Down Trim

Newton's Method Results

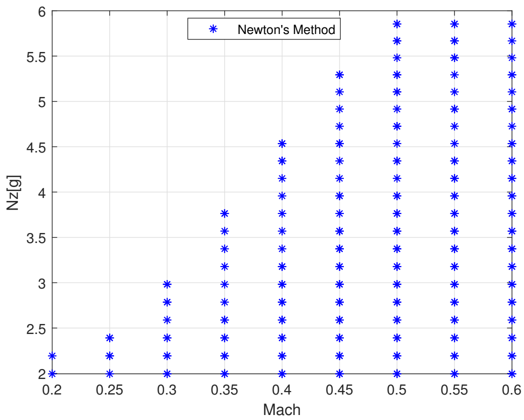
Newton's method is applied first on the fairly simple F16 simulation model for all trim types. Figure 4 (a), (b), (c) shows results of all trim types at various trim points and all the trim points are successfully trimmed.



(a) Wings Level Trims



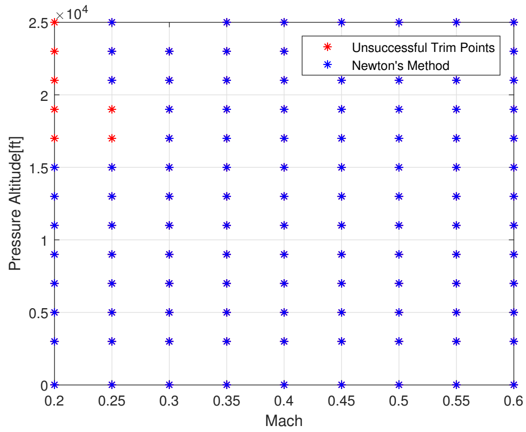
(b) Coordinated Turn Trims



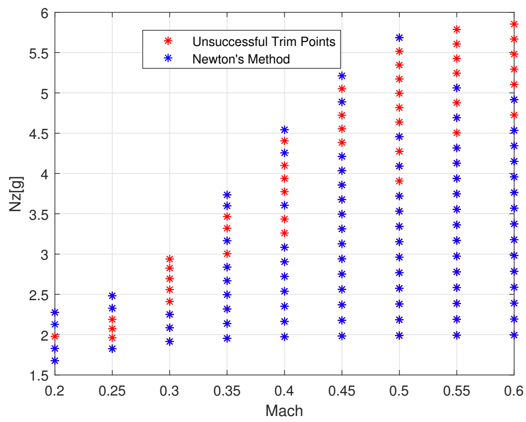
(c) Pull Up Trims

Figure 4: F16 Simulation Model Trims

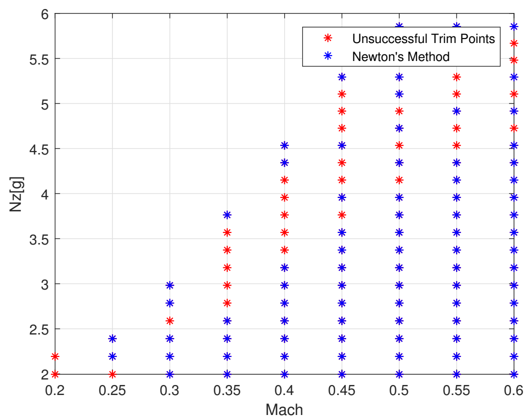
Newton's method is applied also on the Advanced Jet Trainer Aircraft simulation model for all trim types. Figure 5 (a), (b), (c) shows results of wings and level, coordinated turn, and pull up trim conditions, respectively.



(a) Wings Level Trims



(b) Coordinated Turn Trims



(c) Pull Up Trims

Figure 5: Advanced Jet Trainer Simulation Model Trims

As can be seen from the Figure 5, Newton's method degrades in performance for trimming mathematical model which includes high number of nonlinearities. Since aerodynamic nonlinearities becomes more evident at higher angle of attack values, degredation in performance of this method increases. This behavior appears in pull up and coordinated turn trim conditions. Due to the small convergence radius of Newton's method, intermediate points cannot be trimmed. Therefore, in the next part constant adaptive Newton method will be introduced which has a higher convergence radius while conserving quadratic convergence rate.

CONSTANT ADAPTIVE NEWTON METHOD

Convergence radius of Newton's method [Polyak & Tremba , 2019] can be increased by employing an adaptive step size choice. Constant adaptive Newton method is using a simple and naturel way of adapting step size via backtracking (Algorithm 2).

Algorithm 2: Constant Adaptive Newton Method

Result: z
Choose q between [0,1];
Choose initial β_0;
k = 0;
Initialize z_0;
$p_k = \|P(z_k)\|_2$;
while $\|P(z_k)\|_2 > 10^{-8}$ **do**
  $x_k = \underset{P'(z_k)x=P(z_k)}{\operatorname{argmin}} \|x\|_2$;
  $\gamma_k = min\{1, \frac{\beta_k}{p_k}\}$;
  $p_{k+1} = \|P(z_k - \gamma_k x_k)\|_2$;
  while *NOT [*$(\gamma_k < 1\ AND\ p_{k+1} < p_k - \frac{\beta_k}{2})\ OR\ (\gamma_k = 1\ AND\ p_{k+1} < \frac{1}{2\beta_k} p_k)$*]* **do**
    $\beta_k = q\beta_k$;
    $\gamma_k = min\{1, \frac{\beta_k}{p_k}\}$;
    $p_{k+1} = \|P(z_k - \gamma_k x_k)\|_2$;
  end
  $z_{k+1} = z_k - \gamma_k x_k$;
  $\beta_{k+1} = \beta_k$;
  $k + 1$;
end

Moreover, convergence rate is an another important issue with the adaptive Newton methods. If the convergence is linear, iterations can take too long. Beyond extending the convergence radius, this algorithm is also converges quadratically as Newton's method. In Figure 6, both method's convergence histories are compared. At this trim point, Newton's method failed while adaptive one converged to the solution. As can be seen from the convergence histories, adaptive method is linear at first then becomes quadratic.

Another observation from Figure 6, there is a kink which may caused by aerodynamic tables, dead-zones, etc. near by the solution. Due to that kink, Newton's method stuck between two hills and could not pass through to obtain minimum solution because of the large step size. At that kink, adaptive method altered the step size to achieve minimum.

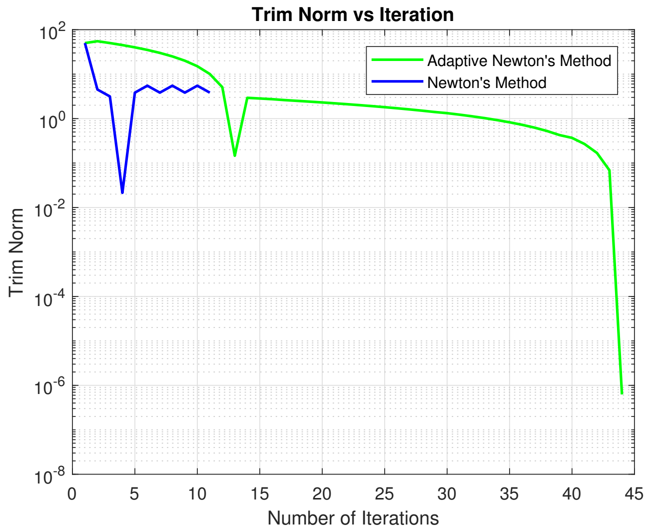
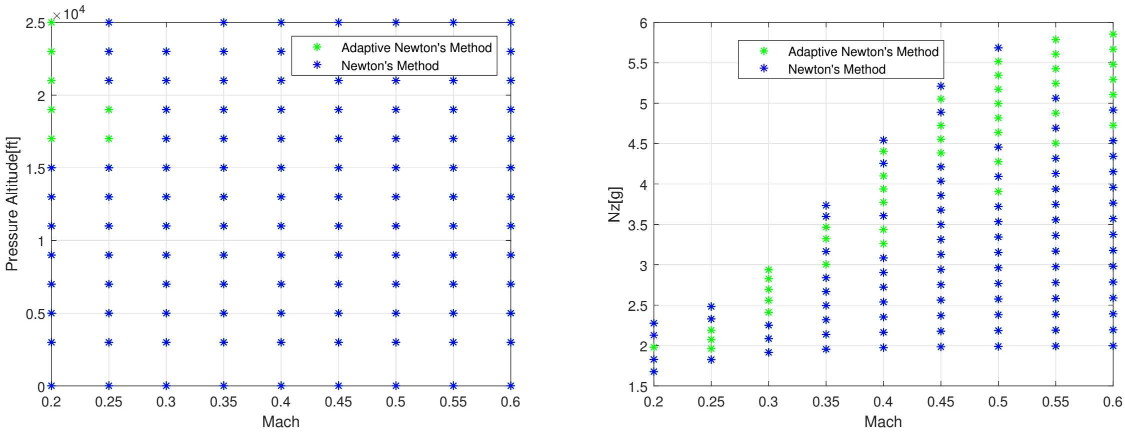
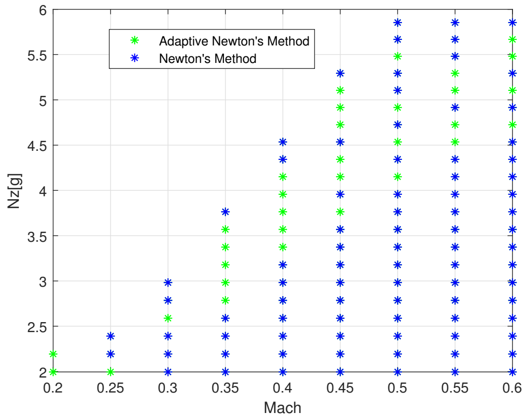


Figure 6: Convergence History Comparison

Constant adaptive Newton method is applied on the Advanced Jet Trainer Aircraft simulation model for all trim types. Figure 7 shows results of wings and level, coordinated turn, and pull up trim conditions, respectively. Trim results cannot be obtained by Newton's method are successfully trimmed by the constant adaptive Newton method.



(a) Wings Level Trims (b) Coordinated Turn Trims



(c) Pull Up Trims

Figure 7: Advanced Jet Trainer Simulation Model Adaptive Method Trims

Since trim operation is a mathematical operation depending on the freeze and float conditons and found trim state and inputs can be another local minimum, simulation results must also be checked to be sure whether trim is true or not. In Figure 8, wings level trimmed states and outputs are given at a random point. As can be seen from figure, given freeze and float conditions are successfully achieved for 5 seconds time interval.

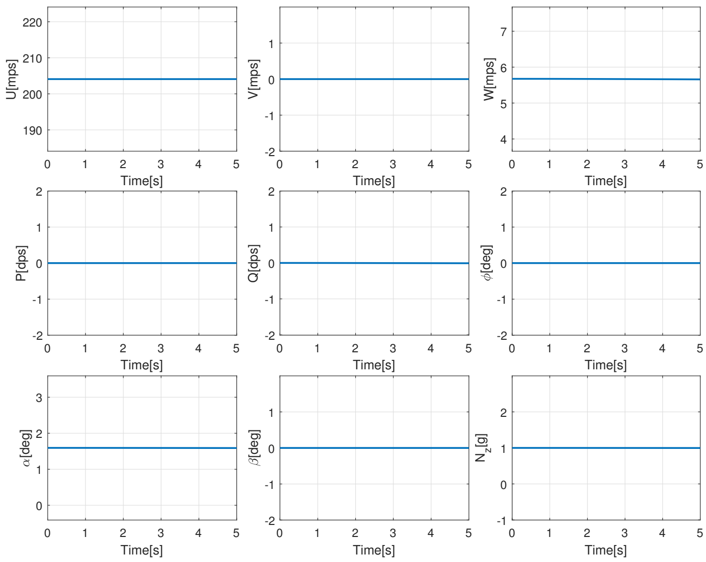


Figure 8: Wings Level Trimmed States and Outputs

In the Figure 9, coordinated turn trimmed states and outputs are presented at a random point. As can be seen from figure, trim values for freeze and float conditions are successfully achieved in five seconds time interval.

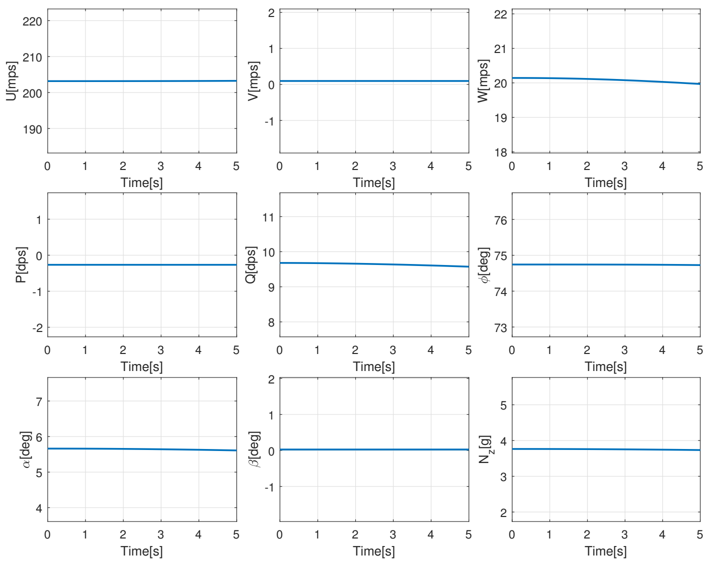


Figure 9: Coordinated Turn Trimmed States and Outputs

In the Figure 10, pull up trimmed states and outputs are presented at a random point. As can be seen from figure, given freeze and float conditions are successfully achieved at beginning of the maneuver, however, since pull up trim condition is a dynamic condition, physically aircraft starts to loose speed due to the conversion of kinetic energy to potential energy while conserving total energy.

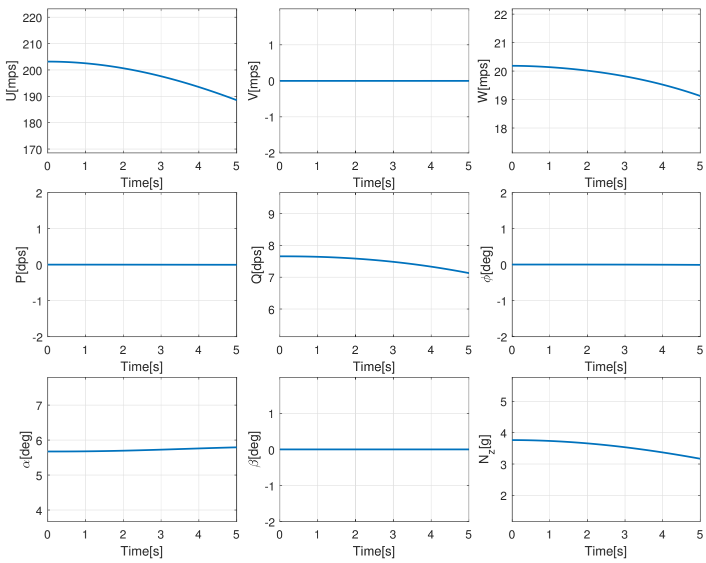


Figure 10: Pull Up Trimmed States and Outputs

CONCLUSION

In this paper, Newton's method and an adaptive scheme called constant adaptive Newton method for aircraft trim problem are compared. Two different aircraft models, F-16 and Advanced Jet Trainer Aircraft, are used to reveal benefits of both methods. It can be concluded that, trimming mathematical models that contain large number of nonlinearities creates a non-convex optimization problem. Therefore, original Newton's method cannot find trim states and inputs, and needs further improvements. Constant adaptive Newton method is an adaptive scheme which alters the step size to solve trim problem. While Newton's method solved trim problem for F-16 simulation model at every trim point, , in the advanced jet trainer simulation model, trim points that cannot be trimmed by Newton's method are trimmed by constant adaptive Newton method. For the further studies, initially chosen parameters of adaptive scheme can also be made adaptive and performance of this algorithm for other trim types can also be explored.

References

Ben-Israel, A. (1966) *A Newton-Raphson method for the solution of systems of equations*, Journal of Mathematical Analysis and Applications, 15(2), 243-252. doi:10.1016/0022-247x(66)90115-6

Huo, Y. (2012) *Model of F-16 Fighter Aircraft*, University of Southern California

Mathworks Inc (2017) *Matlab, Programming Utilities, R2017b*

Nguyen, L. T., Ogburn, M. E., Gilbert,W. P., Kibler,K. S., Brown,P. W., & Deal, P. L. (1979) *Simulator Study of Stall/Post-Stall Characteristics of a Fighter Airplane with Relaxed Longitudinal Static Stability*, (Tech. No. 1538), National Aeronautics and Space Administration

Polyak, B., & Tremba, A. (2019) *Sparse solutions of optimal control via Newton method for under-determined systems*, Journal of Global Optimization, 76(3), 613-623. doi:10.1007/s10898-019-00784-z

Polyak, B., & Tremba, A. (2019) *New versions of Newton method: step-size choice, convergence domain and under-determined equations*, Optimization Methods and Software, 1–32. doi:10.1080/10556788.2019.1669154

Roskam, J. (2001) *Airplane Flight Dynamics and Automatic Flight Controls*, DAR Corp., Lawrance, KS

Stevens, B. L., Lewis, F. L., & Johnson, E. N. (2016) *Aircraft control and simulation: Dynamics, controls design, and autonomous systems*, Hoboken, NJ: John Wiley & Sons