The use of Discrete Rotating Virtual Blades to Suppress the N/Rev oscillations in Flight Mechanics Analyses

Pierluigi Capone (capo@zhaw.ch), Anna Abà (abaa@zhaw.ch)
Zurich University of Applied Sciences, Winterthur, Switzerland

For flight mechanics applications such as handling qualities evaluation, real-time simulation, and flight control system design, a nonlinear rotorcraft simulation model with a reasonably accurate representation of the flapping dynamics is usually considered mandatory. If this is the case, the introduction of the complete flapping dynamics into the model will induce oscillating loads at the rotor hub. This, in turn, creates significant complications in the computation of a correct trim and linearization of the rotorcraft as well as in the execution of stability and control analysis. In the paper, a novel approach to remove such oscillating loads from the nonlinear model, thereby simplifying the trim, linearization, and analysis effort, is presented based on the concept of discrete rotating virtual blades.

Symbols

A, B, C, D	State, Control, Observation, and Feed-through matrices
a_z, a_y	Longitudinal and lateral accelerations [m/s^2]
n	Number of states
N	Number of blades
N_V	Number of virtual blades
p, q, r	Angular rates [rad/s]
Rev	Revolution
u, v, w	Velocity components [m/s]
u, x, y	System inputs, states, outputs
V_T	True airspeed [m/s]
x, y, z	Aircraft position [m]
α, β	Angle of attack and sideslip [rad]
β	Blade flapping angle [rad]
β_{MB}	Blade flapping angle in multi-blade coordinates [rad]
γ	Flight path angle [rad]
τ	Duration of one rotor revolution [sec]
ψ	Rotor Azimuth [rad]
ϕ, θ, ψ	Euler Angles [rad]
Ω	Rotor angular velocity [rad/sec]

Introduction

For flight mechanics applications such as handling qualities evaluations, real-time simulation, and flight control system design, a nonlinear rotorcraft simulation model with a reasonably accurate representation of the flapping dynamics is usually considered mandatory [Padfield-2018]. If this is the case, the introduction of the complete flapping dynamics into the model will induce oscillating loads at the rotor hub.

The transmission of periodic loads to the rotorcraft hub is a known phenomenon [Bielawa-1992], [Guglieri-2018]. Under the assumption of a perfectly balanced rotor, only frequencies multiple of the *N/Rev* frequency (where *N* is the number of blades and *Rev* stands for “revolution”) are passed to the rotorcraft via the hub. While this phenomenon is, of course, crucial for the evaluation of the loading conditions of the rotorcraft, it is less important for the evaluation of flight mechanics characteristics such as handling qualities. The existence of these oscillatory loads creates several problems with regard to the trim and linearization of the aircraft model, as well as the execution of flight mechanics analyses. The paper presents a novel approach that tries to solve these issues using “virtual blades”.

As will be shown in the following, this approach has significant advantages:

1. Trim conditions are no longer affected by oscillating loads. As a result, the exact solution can be found with standard trim algorithms, dramatically reducing convergence time.
2. The proposed approximation has a negligible effect on both the accuracy of the obtained trim conditions and the results of the nonlinear simulation.
3. The matching of the obtained linearized models with the corresponding nonlinear simulations is significantly improved.

The paper is divided into the following parts:

- **Trim & Linearization**: the problem of finding a correct rotorcraft trim condition and the corresponding linearization is discussed.
- **Virtual Blades Approach**: detailed description of the proposed virtual blades method, including a description of the modifications to the simulation model required to implement this novel approach.
- **Accuracy and Computation Loads**: evaluation of computational loads, with the goal of finding the optimal model accuracy for off-line and real-time simulations.

- **Conclusions**: final remarks and conclusions with an outlook on future developments.

The work presented in this paper has been carried out using two rotorcraft models, one of the AW09 and one of the XV15, both developed in-house at ZHAW. A complete model of the AW09 helicopter [Đeverlija-2024] suitable for flight mechanics, control system design, and real-time simulation, is currently under development in the framework of the Innosuisse[1]-funded project WHeSI[2], in collaboration with the helicopter manufacturer Kopter[3]. The multi-purpose mathematical model of the XV15 tilt-rotor research aircraft ([Abà-2023], [Abà-2020], [Barra-2019], [Primatesta-2022], [Primatesta-2023], [Barra-2022]) is used for research activities such as handling qualities studies and flight control systems development.

Trim & Linearization

Because of the oscillatory loads described above, finding the solution of classical rotorcraft trim conditions (such as straight and level, steady turn, steady heading steady sideslip, etc., or the possible variants of these conditions in climb/descent) requires solving a periodic system. Several approaches have been developed by the rotorcraft community to address this issue [Çalişkan-2009], [Demirel-2009], [Flightlab-2023], [Friedman-2008], [Fries-1996], [Padfield-2018], [Peters-1996], and [Sahahat-2017]. At the ZHAW (Zurich University of Applied Sciences), an *ad hoc* method has been developed that is able to determine the exact initial condition (or, actually, a very good approximation of it) of the periodic solution. This method can not only determine the initial mean value of the states, but also their corresponding phasing. Of course, this method is quite expensive in terms of computational power.

A classical method used to solve the trim problem is the Newton-Raphson one. If the simulation model has been correctly built into the simulation environment, the equations of motion can be expressed in the standard form (ordinary differential equations):

$$\begin{aligned} \dot{x} &= f(x,u) \\ y &= g(x,u) \end{aligned} \qquad (1)$$

where u, x, and y are the input, state, and output vectors, respectively. Then, for a steady-state condition (trim condition):

$$\begin{aligned} \dot{x}_0 &= f(x_0,u_0) = const \\ y_0 &= g(x_0,u_0) \end{aligned}$$

If we linearize the equation (1) around the trim condition, we have:

$$\begin{aligned} \Delta\dot{x} &= A\,\Delta x + B\,\Delta u \\ \Delta y &= C\,\Delta x + D\,\Delta u \end{aligned} \qquad (2)$$

where: $\Delta u = u - u_0$, $\Delta x = x - x_0$, $\Delta y = y - y_0$ and $\Delta\dot{x} = \dot{x} - \dot{x}_0$.

Equation (2) can be rewritten as:

$$\begin{Bmatrix} \dot{x} - \dot{x}_0 \\ y - y_0 \end{Bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{Bmatrix} x - x_0 \\ u - u_0 \end{Bmatrix} \qquad (3)$$

The above equation represents an excellent method to compute the initial condition (x_0, u_0). In trim conditions, some of these quantities (u_0, x_0 and $\dot{x}_0$) are known such as, for example, speed and altitude. It is clear that:

- when we "freeze" one element of the state derivative vector to a certain value, we are forcing one row of the right-hand side of this system of linear equations to be equal to this "frozen" value; in other words, we are considering this particular equation for the solution of our problem
- when we "freeze" one element of the output vector, we are doing the same; again, we are considering a particular equation, in this case, an output equation, in the solution of the trim problem
- when we "freeze" one element of the state vector (that means x_j-x_{0j}=0), we are eliminating one column of the right-hand matrix; in practice, we are not considering the corresponding state for the solution of our problem
- when we "freeze" one element of the input vector, we are doing the same; again, we are not considering the corresponding input for the solution of our problem.

Since the inputs and the states that are not "frozen" are the target values that we want to find, the number of these unknowns should be smaller than or equal to the number of equations we are using.

Another way of considering the problem is to still think in terms of "frozen" state derivatives and outputs, but in terms of "floats" (or unknowns) states and inputs. The number of "frozen" (equations = constraints) shall be equal to or greater than the number of "floats" (unknowns). From a numerical point of view, a good rule is always to keep the number of unknowns exactly equal to the number of constraints (equations).

Another important aspect is the initialization of the various unknown values. Since the algorithm performs a linear search, initial values far away from the searched solution may prevent the problem from converging.

[1] Swiss Innovation Agency - www.innosuisse.ch

[2] White-Box Helicopter System Identification

[3] A Leonardo company - www.koptergroup.com

We can now rewrite equation (3) selecting only the rows of the right matrix that correspond to the "frozen" outputs/state derivatives and the columns that correspond to the "floated" inputs/states:

$$\begin{Bmatrix} \dot{x}_{Freeze} - \dot{x}_{0_{Freeze}} \\ y_{Freeze} - y_{0_{Freeze}} \end{Bmatrix} = \begin{bmatrix} A_{Fr\&Fl} & B_{Fr\&Fl} \\ C_{Fr\&Fl} & D_{Fr\&Fl} \end{bmatrix} \begin{Bmatrix} x_{Float} - x_{0_{Float}} \\ u_{Float} - u_{0_{Float}} \end{Bmatrix}$$

Then:

$$\begin{Bmatrix} x_{Float} \\ u_{Float} \end{Bmatrix} = \begin{Bmatrix} x_{0_{Float}} \\ u_{0_{Float}} \end{Bmatrix} + \begin{bmatrix} A_{Fr\&Fl} & B_{Fr\&Fl} \\ C_{Fr\&Fl} & D_{Fr\&Fl} \end{bmatrix}^{-1} \begin{Bmatrix} \dot{x}_{Freeze} - \dot{x}_{0_{Freeze}} \\ y_{Freeze} - y_{0_{Freeze}} \end{Bmatrix} = \begin{Bmatrix} x_{0_{Float}} \\ u_{0_{Float}} \end{Bmatrix} + \begin{bmatrix} A_{Fr\&Fl} & B_{Fr\&Fl} \\ C_{Fr\&Fl} & D_{Fr\&Fl} \end{bmatrix}^{-1} \begin{Bmatrix} \dot{x}_{Freeze} - f_{Freeze}(x_0,u_0) \\ y_{Freeze} - g_{Freeze}(x_0,u_0) \end{Bmatrix}$$

The found input and state variables u_{Float} and x_{Float} can be combined with the remaining frozen input and state variables to form the estimated input and state vectors u_e and x_e. These estimates can be used to evaluate the error e between the searched solution and the obtained approximation:

$$e = \left\| \begin{matrix} \dot{x}_{Freeze} - f_{Freeze}(x_e,u_e) \\ y_{Freeze} - g_{Freeze}(x_e,u_e) \end{matrix} \right\|$$

Using the shown equations, we can build the procedure to trim the aircraft equations of motion. This procedure is summarized in Figure 1.

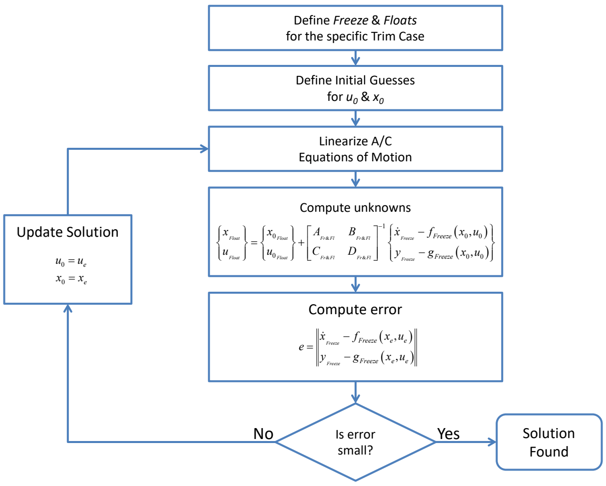

Figure 1 - Trim Procedure Using Newton-Raphson Algorithm

It is worth noting that some trim problems may have multiple solutions. For example, an airplane at a high angle of attack can achieve the same flight condition with different elevator positions and throttle settings. These cases, in general, require ad hoc solutions.

Another problem is trimming close to the minimum or maximum of certain limited variables, such as, for example, control deflections. Since in this case some derivatives may be zero, the algorithm will not be able to find the solution. A very effective workaround is to first trim relatively far from those limits and then use the trimmed inputs and states (u_0 and x_0) as initial value for the required trim condition.

Finally, it must be noted that trim of the simulation model is not only possible in straight and level case, but also for a number of maneuvers for which a particular "steady" state is identifiable. For example, Straight flight in climb or descent, Steady Turn (also in climb or descent), Pull-up / Push-over, Steady Heading Steady Sideslip, Roll at a specified roll rate, etc.

Of course, for this method to work, we must be able to compute the linearization of our model. This is easily done by using numerical perturbations. However, the selection of the inputs and states perturbations may not be straightforward (see [Capone-2002]). The existence of the *N/Rev* oscillations makes this task rather challenging.

When we try to apply the above trim procedure to rotorcraft we encounter two problems. First, we cannot impose conditions on variables to be frozen that are of oscillatory nature. Second, the linearization computed using numerical perturbations will be affected by the periodicity, making the evaluation of the derivatives more difficult and, therefore, having a negative impact on the trim algorithm convergence.

One way to bypass this problem is to evaluate derivatives using a specific time horizon. In steady state conditions, if we swap each blade with the preceding one every $\frac{\tau}{N} = \frac{60}{RPM} \cdot \frac{1}{N}$ sec, we obtain a perfectly repeatable scenario every τ/N sec. Therefore, this seems an appropriate time horizon for the derivatives computation inside the linearization algorithm. Assuming that the time at which we are trimming the aircraft is $t = 0$, the state and output equation generic derivatives of the ODE equation (1) can be expressed as:

$$\left.\frac{\partial f_j(x,u)}{\partial x_i}\right|_{N/\mathrm{Rev}} \approx \frac{f_j(\Delta x_i,u) - f_j(x(0),u(0))}{x_i(\tau/N) - x_i(0)}$$

$$\left.\frac{\partial g_j(x,u)}{\partial x_i}\right|_{N/\mathrm{Rev}} \approx \frac{g_j(\Delta x_i,u) - g_j(x(0),u(0))}{x_i(\tau/N) - x_i(0)}$$

where:

$$x = \{x_1(0) \quad x_2(0) \quad \cdots \quad x_j(0) \quad \cdots \quad x_n(0)\}^T$$
$$\Delta x_i = \{x_1(0) \quad x_2(0) \quad \cdots \quad x_i(\tau/N) \quad \cdots \quad x_n(0)\}^T$$

In this case, the computation of the derivatives requires several simulations (as many as there are "free" states and inputs) of the whole rotorcraft model for τ/N seconds. This is, of course, quite demanding from the computational point of view.

For a rotorcraft, the trim condition is not only given by the average values of the states and outputs but also by the relative phasing. We need a method that not only determines the initial mean value of the states, but also their corresponding phasing. This can be achieved by adjusting the state values $\overline{x}$ computed at each iteration of the trim algorithm as follows:

$$\overline{x} = \{\overline{x}_1 \quad \overline{x}_2 \quad \cdots \quad \overline{x}_j \quad \cdots \quad \overline{x}_n\}^T$$
$$\overline{x}_j^{Adjusted} = x_{Freeze\,j} + x_j(\tau/N \mid x(0) = \overline{x}) - \frac{1}{\tau/N}\int_0^{\tau/N} x_j(t \mid x(0) = \overline{x})\,dt$$

The recovery of the phasing between the different state variables and its positive effects on trim are shown in Figure 2 with regard to the angular rates.

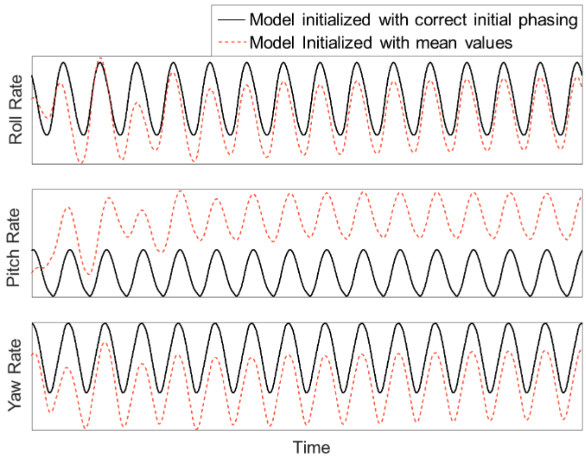

Figure 2 – Comparison of different simulation initializations (angular rates)

As mentioned, if the initial condition is not calculated accurately, the various techniques for numerical computation of the linear model can easily fail. For example, let's consider a trim condition where the initial conditions were not precisely achieved. Figure 3 shows the comparison of the finite-horizon linearization with the classical one for this "non precise" trim. While the finite-horizon linearization is able to catch the correct dynamics of the flight mechanics states, it fails to catch the high frequency flapping dynamics. On the other hand, the classical linearization correctly represents the flapping dynamics, but not the flight mechanics modes. More insight into this problem can be found in [Barra-2022], [Capone-2002], and [Dreier-2018].

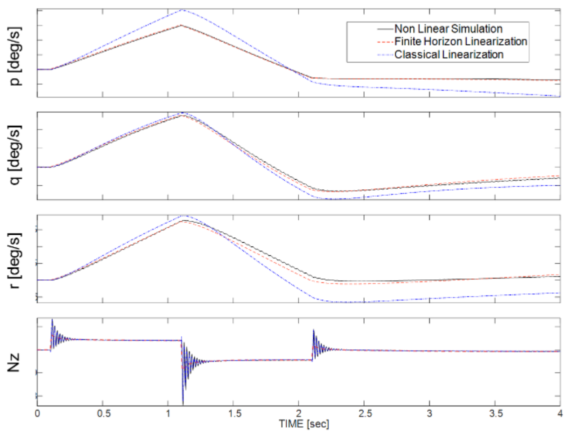

Figure 3 – Comparison of different linearization techniques (XV15 Model at 40 kts)

It must also be mentioned that additional complications exist if the rotorcraft simulation model is realized in discrete-time domain, as is usually the case for models that must run in real-time. For example, the τ/N time may not be a multiple of the chosen sample time. In this case, the computation of the finite-horizon derivatives as well as the evaluation of the phasing of the initial states requires additional ad hoc methods.

Virtual Blades Method

As shown in the previous sections, the existence of oscillations at the *N*/*Rev* frequency complicates flight mechanics analysis. Nevertheless, these oscillations are the observable effect of a real phenomenon and cannot be simply ignored.

An innovative approach has been taken to suppress these oscillations in steady conditions such as trim and to significantly reduce them during maneuvering. In this approach, the rotating blades are replaced by virtual ones. The term "virtual" is used in the sense that:

- The number of virtual blades N_V does not need to be the real one; this has to be equal to or greater than the true number of blades N.
- These virtual blades are subjected to all inertial (including centrifugal), aerodynamic, and structural forces and moments, but they occupy fixed azimuth positions on the rotor disk.

In fact, once the number of virtual blades N_V is selected, each virtual blade is kept in one of these N_V azimuth positions for a time corresponding to $\frac{\tau}{N_V} = \frac{60}{RPM \cdot N_V}$ seconds, while the rest of the kinematic variables are correctly computed. After $\frac{\tau}{N_V}$ seconds, the 1st blade is replaced by the N_v^{th}, the 2nd by the 1st, the 3rd by the 2nd, and so on.

From a modeling point of view, the implementation of the Virtual Blades is quite simple, given that the modeling of the rotor itself has been done in a flexible and parametric way. The ZHAW rotor model is implemented in the Matlab/Simulink® simulation environment. The model has been developed over the years and is fully parametrized, for example, with respect to the number of blades. Therefore, changing this number is just a matter of modifying the corresponding input text file.

The model was subsequently modified introducing, alongside the real number of blades N, the number of virtual blades N_V. Of course, all the rotor-generated forces and moments must be scaled down by the factor N/N_V.

The second step was to discretize the azimuth value fed to the flapping dynamics as per Figure 4. This discretization affects only the flapping dynamics but not at all the variables that feed it such as, for example, the angular rates that come from the equations of motion.

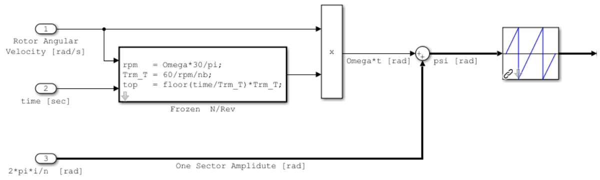

Figure 4 – Azimuth Angle Discretization

In this process, the degrees of freedom of the blades are not altered. The real $2 \cdot N$ blade states (flapping angles and their derivatives) are expressed in terms of multi-blade coordinates β_{MB} and then transformed into the local virtual blade $2 \cdot N_V$ flapping angles using the multi-blade transformation. Once the N_v local flapping derivatives have been computed, these are transformed back into the $2 \cdot N$ multi-blade coordinates. The following equations clarify the process for the case $N = 5$:

$$\Psi = \{\psi_1 \quad \psi_2 \quad \dots \quad \psi_{N_V}\}^T$$
$$M_{MB2BL} = [I \quad \sin(\Psi) \quad \cos(\Psi) \quad \sin(2 \cdot \Psi) \quad \cos(2 \cdot \Psi)]$$
$$\dot{M}_{MB2BL} = \Omega \cdot [0 \quad \cos(\Psi) \quad -\sin(\Psi) \quad 2\cos(2 \cdot \Psi) \quad -2\sin(2 \cdot \Psi)]$$
$$\beta = M_{MB2BL}\beta_{MB}$$
$$\dot{\beta} = M_{MB2BL}\dot{\beta}_{MB} + \dot{M}_{MB2BL}\beta_{MB}$$

then, once the updated flapping dynamic states $\dot{\beta}$ and β have been computed:

$$M_{BL2MB} = \frac{1}{N_V}[I \quad 2\sin(\Psi) \quad 2\cos(\Psi) \quad 2\sin(2 \cdot \Psi) \quad 2\cos(2 \cdot \Psi)]^T$$
$$\dot{M}_{BL2MB} = \frac{\Omega}{N_V}[0 \quad 2\cos(\Psi) \quad -2\sin(\Psi) \quad 4\cos(2 \cdot \Psi) \quad -4\sin(2 \cdot \Psi)]^T$$
$$\beta_{MB} = M_{BL2MB}\beta$$
$$\dot{\beta}_{MB} = M_{BL2MB}\dot{\beta} + \dot{M}_{BL2MB}\beta$$

where the size of the matrix M_{BL2MB} is $N_V \times N$ ($N_V \times 5$) and the size of M_{MB2BL} is $N \times N_V$ ($5 \times N_V$).

From this, it follows that the number of virtual blades can be arbitrary, but on the other hand, some physical considerations need to be made. First, it does not make sense to have fewer virtual blades than the real ones. Second, for a rotor having 2 blades, the method will not be able to generate the correct forces and moments for $N_V = 2$. Third, for a rotor with 3 blades $N_V = 3$ may not be sufficient to represent correctly the rotor forces and moments. This is, for instance, the case of the XV15 as it will be shown in the following.

With the above approach, all the dependency from the azimuth is frozen in the sector that corresponds to a specific virtual blade. Hence, all the forces and moments generated by any virtual blade while this is in the corresponding sector for a time of $\frac{\tau}{N_V}$ will depend on a fixed azimuth value. Therefore, in steady state conditions (trim conditions), all forces and moments generated will be constant for $\frac{\tau}{N_V}$ seconds in every sector. The scenario is portrayed in Figure 5 for the case $N_V = 5$. As can be seen, the behavior of a generic force/moment is still periodic of period $1/\tau$, but if we now iteratively shift the 1st virtual blade with the 5th and the 2nd with the 3rd, and so on, the total sum of the force/moment value does not change. Therefore, we achieved that in a steady-state condition, the generic resulting force/moment at the hub is constant along a whole rotation.

Figure 5 – Generic Force / Moments behavior in Steady-State condition with the Virtual Blade approach with $N_V = 5$

As a final remark, it should be noted that, because of the simplicity of the virtual blades method, a simulation model can be built in such a way that it can run both with real blades or with a generic number of virtual blades. All the rotorcraft models developed at the ZHAW possess this capability.

Accuracy and Computational Loads

As anticipated, we cannot set the number of virtual blades to any number. This must be consistent with a reasonable approximation of the forces and moments acting at the hub. For example, in the case of XV15, it is not possible to use only 3 virtual blades, which is the actual number of blades. Figure 6 compares the XV15 trim results, obtained using the original simulation model ("Base Model") and the virtual blade approach for N_V equal to 3 and 6.

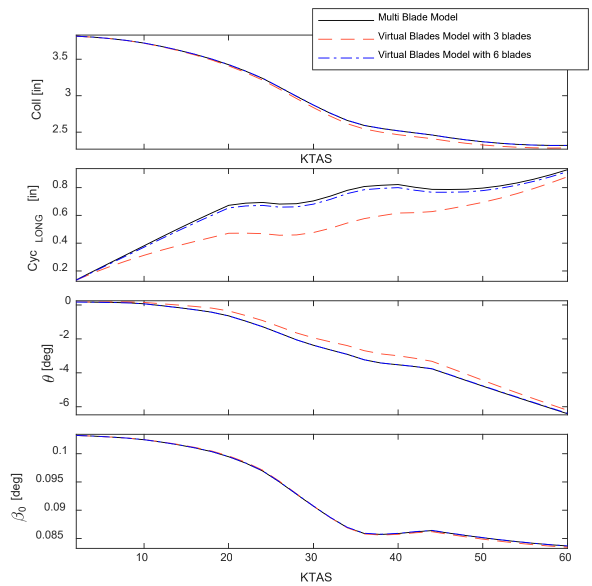

Figure 6 – Trim results comparison (XV15 Model)

As can be seen, a small number of virtual blades is sufficient to provide more than reasonable results. While $N_V = N \cdot 2 = 6$ is necessary for the XV15 model, for a model like the AW09 $N_V = N = 5$ is already appropriate, as shown in Figure 7, where no improvement is noticeable by setting $N_V = 10$.

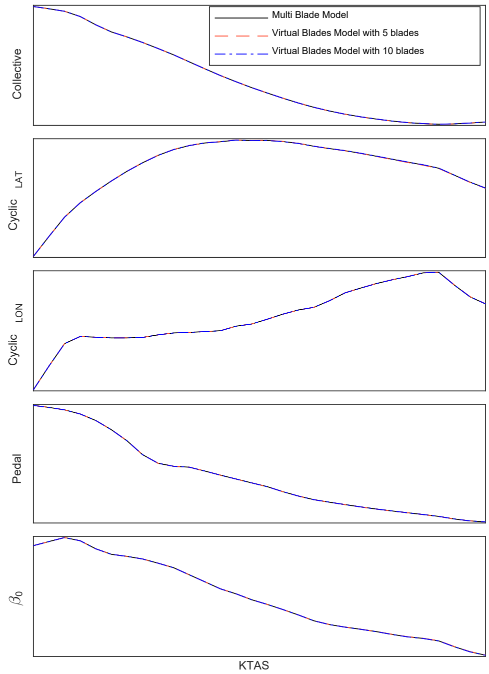

Figure 7 – Trim results comparison (AW09 Model)

Figure 8 and Figure 9 highlight the dramatic reduction in computational time when using the virtual blades method for XV15 and for the AW09, respectively.

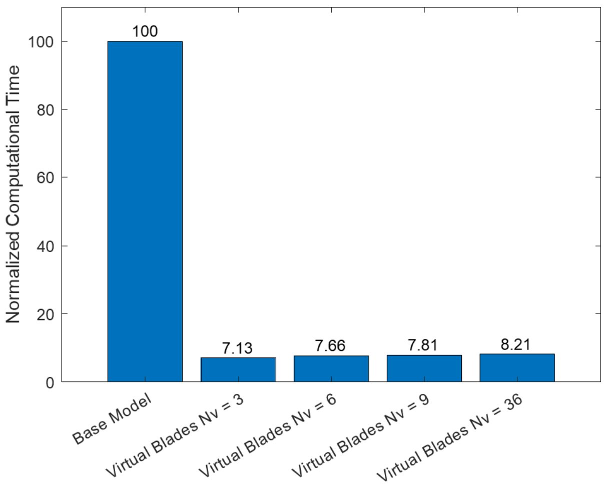

Figure 8 – Trim computational load comparison (XV15 Model)

An interesting aspect is the small increase in computational load as the number of virtual blades, N_V, increases, at least for trim computation. In fact, since a standard computer running Matlab/Simulink® was used to evaluate the computational load, given the small differences in terms of computational load between the $N_V = 5$, 10, and 15 cases, the resulting load for $N_V = 5$ was actually higher than the ones for $N_V = 10$ and 15.

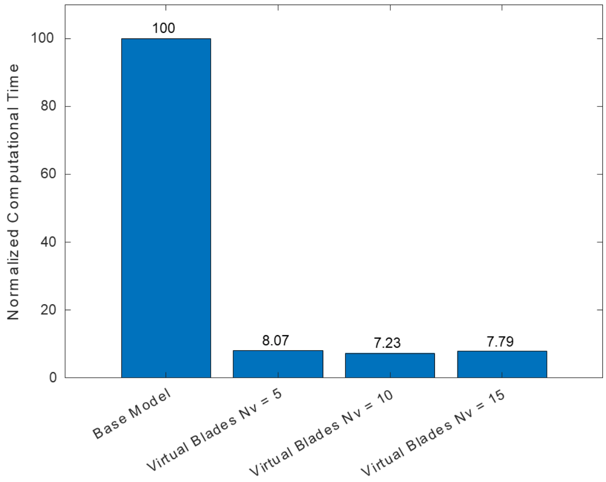

Figure 9 – Trim computational load comparison (AW09 Model)

The accuracy of the virtual blades method can also be verified by comparing the results of nonlinear simulations using the two different approaches. For example, Figure 10 shows the comparison of cyclic doublets response of the AW09 for $N_V = 5$,10, and 15. As can be seen, the differences are negligible. The corresponding normalized errors are reported in Table 1, where the normalized error is computed as:

$$NormErr = \frac{\sum_i \left| y_i^{Multi\ Blades} - y_i^{Virtual\ Blades} \right|}{Number\ of\ Samples} \cdot \frac{1}{\max\left(\left| y^{Multi\ Blades} \right| \right)} \cdot 100$$

	p	***q***	***r***	***β₀***	***az***
$N_V = 5$	0.31	0.72	0.84	0.08	0.09
$N_V = 10$	0.12	0.13	0.31	0.03	0.03
$N_V = 15$	0.05	0.15	0.11	0.03	0.03

Table 1 - Simulation normalized error, cyclic doublets (AW09 Model)

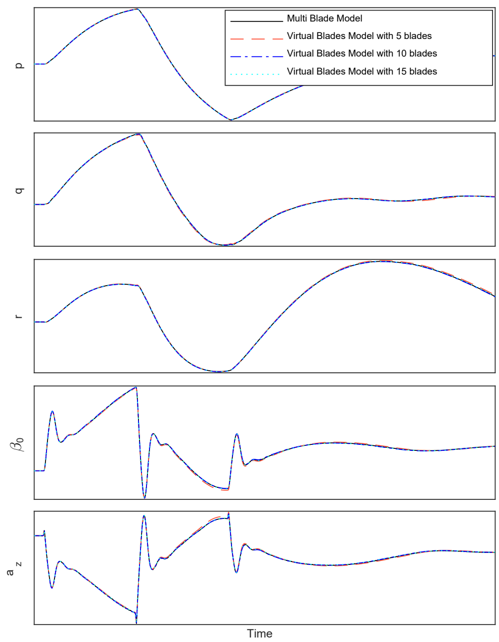

Figure 10 – Simulation comparison, cyclic doublets (AW09 Model)

To demonstrate the effectiveness of the method in suppressing oscillatory loads at the hub, a collective doublet was applied to the AW09 model, and a frequency analysis was carried out on the acceleration output for the two cases: using the real blades and using the virtual blades with $N_V = N = 5$. The results are shown in Figure 11. The effectiveness of the virtual blades in attenuating the N/Rev frequencies, without significantly affecting the nonlinear flight mechanics response, is evident.

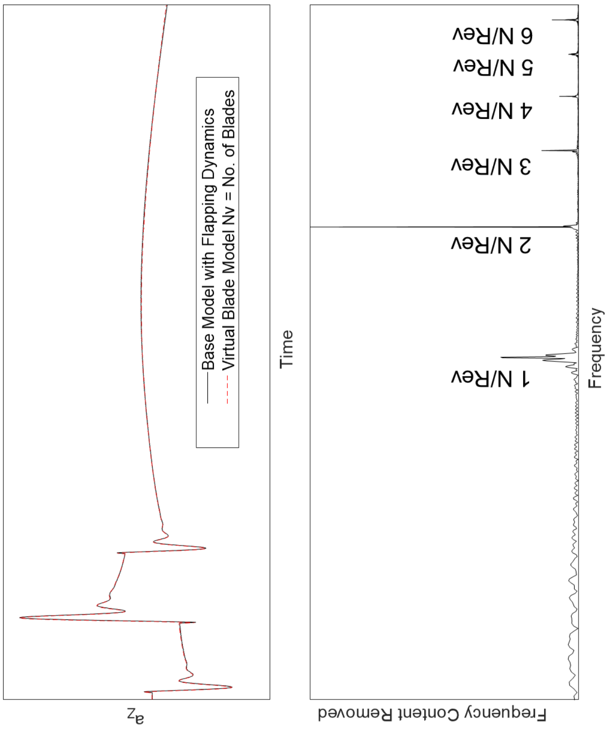

Figure 11 – Frequency Analysis of Oscillating Loads (AW09 Model)

Conclusions

The introduction of the complete flapping dynamics into a rotorcraft simulation model induces oscillating loads at the rotor hub. This, in turn, creates significant complications in the computation of a correct trim and linearization of the rotorcraft as well as in the execution of stability and control analysis.

In the paper, a novel approach to remove such oscillating loads from the nonlinear model, thereby simplifying the trim, linearization, and analysis effort, is presented based on the concept of discrete rotating virtual blades.

The effectiveness and accuracy of the method have been demonstrated with comparison with the correct/classical modeling approach of two rotorcraft models, the AW09 and the XV15. Additionally, it has been shown that the computational efficiency of the method allows for a reduction in computation time by more than an order of magnitude.

In the near future, the virtual blades method will be used for the parameter estimation of the AW09 helicopter, as part of the Innosuisse-funded project WHeSI in collaboration with Kopter.

Acknowledgements

The authors want to thank Dr. Francesco Scorcelletti, Mr. Alessandro Masi and Dr. Matteo Daniele from Kopter Group - a Leonardo Company - for their support and invaluable feedback on this work.

References

[Abà-2020] Mathematical Modelling of Gimballed Tilt-Rotors for Real-Time Flight Simulation, A. Abà, F. Barra, P. Capone, G. Guglieri, MDPI Aerospace Open Access Journal, 2020

[Abà-2023] Convertiplane Control During Transition: the Collective and Throttle Lever (CATL), A. Abà, A. Pedrioli, R. Monstein, M. Hitz, G. Schneider, P. Capone, European Rotorcraft Forum 2023

[Barra-2019] Implementation of a Comprehensive Mathematical Model for Tilt-Rotor Real-Time Flight Simulation, F. Barra, S. Godio, G. Guglieri, R. Mon-stein, P. Capone, 45th European Rotorcraft Forum, Warsaw, Poland, September 2019

[Barra-2022] Doctoral Dissertation, Mathematical Modelling of Tilt-Rotor Aircraft Configurations, A comprehensive model for flight control system development and real-time piloted simulation, Politecnico di Torino, F. Barra, 2022

[Bielawa-1992] Rotary Wing Structural Dynamics and Aeroelasticity, Bielawa R.L., AIAA, 1992

[Çalişkan-2009] Development of Forward Flight Trim and Longitudinal Dynamic Stability Codes and Their Application to a UH-60 Helicopter, S. Çalişkan, Thesis, Middle East Technical University, 2009

[Capone-2002] Linearization of aircraft models: a flight control system and flying qualities perspective. In Challenges in dynamics, system identification, control and handling qualities for land, air sea and space vehicles. RTO Systems - Concepts and Integration Panel (SCI) Symposium, P. Capone, 2002

[Demirel-2009] Investigating Trim Algorithms for a Utility Helicopter, Anil Demirel, Thesis, Middle East Technical University, 2009

[Đeverlija-2024] A Novel Wake Interference Model for Helicopter Real-Time Simulation, J. J. Đeverlija, A. Abà, P. Capone, A. Masi, M. Daniele, F. Scorcelletti, European Rotorcraft Forum 2024

[Dreier-2018] Introduction to Helicopter and Tiltrotor Flight Simulation. Second Edition. American Institute of Aeronautics and Astronautics Inc., M. Dreier, 2018

[Flightlab-2023] Flightlab X-Analysis User Manual, Advanced Rotorcraft Technology, Inc., 2023

[Friedman-2008] A Highly Robust Trim Procedure for Rotorcraft Simulations, Chen Friedman and

Omri Rand, AIAA Modeling and Simulation Technologies Conference and Exhibit, 2008 https://doi.org/10.2514/6.2008-6361

[Fries-1996] Helicopter Trim Analysis, Joseph Fries, ARMY RESEARCH LABORATORY, ARL-MR-317, 1996

[Guglieri-2018] Meccanica del Volo Dell'Elicottero, G. Guglieri, M. Porta, A. Quinci, Esculapio, 2018

[Lyu-2019] Method for eliminating aerodynamic lift vibration of rigid rotor helicopters based on the novel sine-trim model, Wei-Liang Lyu, Aerospace Science and Technology, 2019, https://doi.org/10.1016/j.ast.2019.105655

[Lyu-2020] Improved sine-trim^2 model for the simultaneous elimination of aerodynamic lift, roll, and pitch moment vibrations of rigid rotor helicopters, Wei-Liang Lyu, Aerospace Science and Technology, 2020, https://doi.org/10.1016/j.ast.2020.106235

[Padfield-2018] Helicopter Flight Dynamics, Including a Treatment of Tiltrotor Aircraft, Third Edition, Gareth D. Padfield, Wiley, 2018

[Peters-1996] A General Theory of Rotorcraft Trim, David A. Peters and Dinesh Barwey, MPE - Volume 2, pp. 1-34, Overseas Publishers Association, 1996

[Primatesta-2022] 2022 Design and Integration of a Tilt-Rotor Flight Simulation Platform, S. Primatesta, F. Barra, P. Capone, G. Guglieri, Vertical Flight Society's 78th Annual Forum & Technology Display, Ft. Worth, TX, USA, 2022

[Primatesta-2023] 2023 Implementation of a Comprehensive Real-Time Flight Simulator for XV-15 Tilt-Rotor Aircraft, S. Primatesta, F. Barra, S. Godio, G. Guglieri and P. Capone, AIAA SciTech Forum, 2023

[Sahahat-2017] Modeling, Trimming and Simulation of a Full Scale Helicopter, Hassan Shahat Hassan, Amgad M. Bayoumy, Gamal M. El-Bayoumi, Mohamed Madbouly Abdelrahman, 17th International Conference on Aerospace Sciences & Aviation Technology, ASAT-17-016-MO, 2017