JETTISONING SIMULATION IMPROVEMENTS
–
PROCESS, MODELING AND VALIDATION WITH FLIGHT TEST RESULTS

T. Ries, U. Kiesewetter
Flight-Mechanics
Eurocopter Deutschland GmbH
86607 Donauwoerth, Germany
e-mail: tobias.ries@eurocopter.com, uwe.kiesewetter@eurocopter.com

Abstract

In the paper the recent achievements in improving the Jettisoning-Simulation-Environment used at EC (Eurocopter) are presented. The simulation environment which is based on 6DoF-Flight-Mechanical-Simulation-Tools in combination with CFD-Calculations and Collision detection tools was recently improved with respect to the process, the physical modeling and by additional validations with flight tests. The setup process was completely revised with respect to serialization and the calculation process was accelerated by parallelization. The physical modeling was improved by introducing an interaction model between the H/C and the External and by including a new collision detection method which takes the exact geometries of the bodies into account. In order to show the operability of the method, a comparison between the simulation and a real jettisoning case is provided.

ABBREVIATIONS

6DoF	6 Degree of Freedom
c.g.	center of gravity
CoS	Coordinate-System
EC	Eurocopter
H/C	Helicopter
PyJAMA	Python Jettisoning and Armament Monitoring Application
PiTwice	Polar Import, Transformation, Weighted Interpolation, Conversion and Export

NOMENCLATURE

α	Angle of attack
β	Sideslip angle
φ	Roll angle
θ	Pitch angle
ψ	Yaw angle
Ω	Absolute rotational velocity with $\Omega = \lvert\vec{\omega}\rvert_2$
$\vec{\omega}$	Rotational velocity vector $\vec{\omega} = (\omega_x, \omega_y, \omega_z)$
A	Altitude
E	Energy
f	Function
I	Largest element of moment of inertia tensor
m	Mass
N	Count
p	Roll rate $p = \dot{\varphi}$
q	Pitch rate $q = \dot{\theta}$
r	Yaw rate $r = \dot{\psi}$
$\vec{r}_{A-B}$	Distance vector between point A and B
R	Rate of Descent
S	Sideslip angle
t	Time
T	Temperature
$\vec{v}$	Velocity vector $\vec{v} = (v_x, v_y, v_z)$
$\vec{v}_i$	Main rotor induced downwash velocity vector $\vec{v}_i = (v_{ix}, v_{iy}, v_{iz})$
V	Absolute velocity with $V = \lvert\vec{v}\rvert_2$
x	x-coordinate,
y	y-coordinate
z	z-coordinate

$_{HC}$	H/C-fixed, flight-mechanical CoS
$_{g}$	Ground fixed coordinate system
$_{Ext}$	External Jettisoning Body
$_{rel}$	Relative value
$_{rot}$	Rotational
$_{cg}$	Center of Gravity
$_{colp}$	Collision point
$\lvert\ \rvert_2$	Euclidian norm

1 INTRODUCTION

Jettisoning of external bodies in emergency situations is a crucial part in the certification process of new external bodies like e.g. fuel tanks for a helicopter. An envelope which defines the flight states which are safe for jettisoning of an external body needs to be created for each external body of a helicopter. Jettisoning flight tests are expensive and can only cover single flight states while the proof for certification needs to cover whole flight envelopes with several variables, resulting in a test matrix of thousands of individual flight states. Therefore a jettisoning simulation environment is in use at EC to prove the jettisoning-safety for whole flight envelopes. The generation of the envelopes is based on three steps: 1. Database generation for step two and three, 2. Trajectory simulation of the H/C and the jettisoned external body, 3. Collision detection between H/C and jettisoned external body. In the recent past the demand and accuracy requirements for such simulations increased significantly, so that the whole environment was improved with respect to all parts the process. The database generation process (1.) was completely revised with respect to serialization and higher accuracy. In the trajectory simulation process (2.), the physical modeling was improved by introducing an interaction model between the H/C and the external body as well as an improved serialization. Finally the collision detection process (3.) was enhanced by including a new collision detection method which takes the exact geometries of the bodies into account and the parallelization of the whole process for distributed calculation on multiple CPUs. In this paper the improvements are presented in detail together with a comparison of a real jettisoning flight test with results of the simulation environment.

2 JETISONING SIMULATION PROCESS

The generation of a flight envelope for safe jettisoning is divided into the following three simulation steps:

1. Database generation for step two and three and setup of the case, consisting of:

External Body:

- CAD-Generation of simplified external body
- CFD-Calculation of external body for different α/β inflow angle combinations
- Generation of a smooth aerodynamic α/β-dependent force- and moment-polar from CFD-simulation results
- Setup of the kinematic model of the external body with aerodynamic polar, mass, c.g. (center of gravity) and moments of inertia.

Helicopter:

- CAD-Generation of simplified H/C-fuselage
- CFD-Calculation of H/C-fuselage for different α/β inflow angle combinations
- Extraction of the flow vectors at the position of the external from the CFD-Calculations
- Generation of a smooth α/β-dependent difference velocity vector polar
- If not already existent, generation of the H/C dataset for flight mechanical trajectory simulations

2. Trajectory simulation of the H/C and the jettisoned external body

This step is based on 6DoF-flight-mechanical H/C simulation tools. The tools are the companies standard tools used in daily development for flight mechanical simulations and are capable to calculate time dependent trajectories of helicopters and, if required, other bodies like for example jettisoned external bodies. The simulation consists of the following parts:

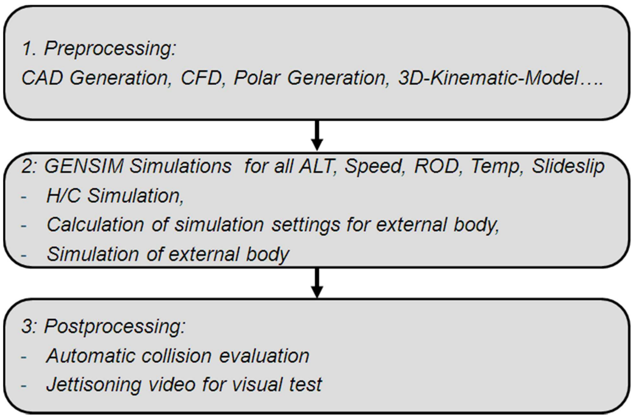
- Creating the flight-state-combination-lists for:
    - Total inflow velocity (V)
    - Rate of Descent (R)
    - Sideslip angle (S)
    - Temperature (T)
    - Altitude (A)

- Fully automated calculation procedure: It calculates the trimmed H/C trajectory and the jettisoning trajectory of the external body with the according trimmed H/C-state as initial boundary conditions for all possible combinations of the input values provided in the lists.

3. Collision detection between H/C and jettisoned external body, consisting of:

- Fully automated collision detection by evaluation of trajectories in combination with 3D-CAD-Files of Helicopter and external body for each simulated time-step.
- Evaluation results for each flight state are summarized in one file suitable for the post-processing for an easy envelope creation.
- For each simulated flight state a 3D-File suitable for post-processing can be created which includes the whole jettisoning sequence. This can be used to visualize the jettisoning process, to compare it with flight test recordings and to create videos.

In Figure 1 a summary of the whole jettisoning process is provided.



1. Preprocessing:
CAD Generation, CFD, Polar Generation, 3D-Kinematic-Model….

2: GENSIM Simulations for all ALT, Speed, ROD, Temp, Slideslip
- *H/C Simulation,*
- *Calculation of simulation settings for external body,*
- *Simulation of external body*

3: Postprocessing:
- *Automatic collision evaluation*
- *Jettisoning video for visual test*

Figure 1: Overview of the whole Jettisoning process

3 PROCESS IMPROVEMENTS

Recently the each of the three steps of the jettisoning process was significantly improved. The improvements got necessary because in the recent past, the number of required jettisoning simulations increased significantly. In order to be able to provide reliable and comparable simulation results in limited time, the whole process was revised with the focus on:

- Reproducibility
- Reduction of work steps
- Automation of error prone work steps
- Speed up of simulations

The most effort for step one was put in the serialization of the aerodynamic polar generation. The most important data for a reliable aerodynamic simulation are the pre-calculated spherical polars. They provide the aerodynamic forces and moments for a body in all three directions in space for every possible incident angle combination of α and β to the flight mechanical tools which generate the trajectories. They cover all possible inflow directions with $\alpha \in [-90°;+90°]$ and $\beta \in [-180°;+180°]$. In the past these polars were generated by a manual process in which the CFD-calculations, the coordinate-system transformations, the interpolation between the simulated CFD-inflow-angle-combinations and the output-formatting suitable for the flight mechanical tools were done in manual, individual steps. Now, a standard CFD-setup is available which just requires a grid of the external body with a sufficient far-field and a list of α and β combinations which need to be calculated. Usually a fixed list of about 40 α / β combinations is calculated. The CFD-calculation process is fully automated and comprises the creation of the individual working directories, setting of boundary conditions, launching the jobs in parallel on the simulation-cluster and collection of the required results. These results can directly be used to create the aerodynamic polar in the input format which is required for the flight mechanical simulation tools. A program named $\pi \times II$ (Pi-Twice, Polar Import, Transformation, Weighted Interpolation, Conversion and Export) was developed at EC, which first automatically reads all force and moment outputs located in a given directory including the corresponding inflow angle combinations. Then the forces and moments get averaged over a defined number of time steps. With this information the forces and moments get interpolated in to a defined matrix of α / β combinations which represent the spherical polar. All connection conditions of the polar edges, like $f(\alpha=+90°, \beta=0°) = f(\alpha=-90°, \beta=180°)$, are taken into account by the program. Different interpolation methods can be chosen, from simple linear interpolation to sophisticated Kriging or Radial Basis Function interpolation. The program is also able to deal with any arbitrary list of α / β combinations, so that the CFD-Simulations of Wind Tunnel tests which provide the input data do not need to fulfill any special specifications for the values of the angles. It is also possible to make use of plane symmetries, so that for symmetric bodies the number of required CFD-Simulations is significantly reduced. The program can either be run by a GUI or in batch mode and can also just calculate individual interpolation values instead of whole matrices. The output format can be set to the required input format for the flight mechanical tools and all required coordinate transformations are performed automatically. The polar generation is usually done with standard settings which allow a direct conversion of the CFD-Output to an aerodynamic force and moment polar in the required input format and coordinate system of the flight mechanical tool. This procedure dramatically reduced the time required for a polar generation from the original geometry to the final polar and eliminates many error prone working steps.

For simulation step two, the trajectory calculation was significantly simplified, by merging all settings in one configuration file and automation of the file-generation required for the new fuselage-influence-model. Also a new calculation series management was introduced, which allows for a quick and simple way to create multiple calculation series with a single setup.

In simulation step three the collision detection routine was replaced by a more accurate program, which in return requires orders of magnitudes more computational time. This was also the reason for taking the evaluation process out of the serial calculation procedure of the trajectories as it was done up to date. Now the trajectories of all cases are calculated and stored first and subsequently the collision evaluation is computed based on this database. Besides the only negative fact, that for this procedure more storage space is required, it has two significant advantages. Firstly the evaluation can easily be parallelized since the evaluations of the individual cases are independent from each other and secondly jettisoning simulations of different helicopter configurations can be performed with the same trajectory database by just changing the 3D-Helicopterfile which is used for the collision detection. However for such different helicopter configurations it must of course be ensured, that the effect of the configuration change on the flow field especially in the vicinity of mounting position of the external body is negligible. Also the collision detection process is fully automated and compatible to the calculations series management of the trajectory generation process. The collision detection simulations are distributed automatically on an arbitrary number of CPUs and the results are collected and combined to a single result file when all parallel processes are finished. Depending on the complexity of the 3D-Bodies this allows to evaluate a typical number of about 1300 jettisoning flight states within down to less than an hour instead of several days with serial processing.

4 PHYSICAL MODELING IMPROVEMENTS

With regard to the physical modeling the jettisoning simulation process was improved in three points:

- Representation of the aerodynamic influence of the fuselage on the jettisoning external body
- Accuracy of the collision detection simulation
- Determination of collision energies

4.1 Aerodynamic influence of the fuselage

A significant problem for the accuracy of jettisoning simulations by polar and blade element based simulation tools is the fact that usually the helicopter and the external are simulated in two separate independent simulations. This leads to the problem that the aerodynamic influence of the helicopter on the jettisoned external body is not or not well represented in the simulations. First the helicopter trajectory is generated and then the boundary conditions of the simulation of the external body are set by results of the trimmed helicopter at the start time of the jettisoning. This comprises the velocity vector over ground $\vec{v}_g = (v_{xg}, v_{yg}, v_{zg})$, the attitude angles φ, θ, ψ and the turn rates p, q, r. Also the influence of the main rotor induced downwash velocity $\vec{v}_i$ was already taken into account. However, except of the downwash velocity, no aerodynamic interaction between the helicopter and the external body was taken into account in the past despite the fact, that these interactions can significantly influence the simulation results. In order to improve the accuracy of the simulation with regard to aerodynamic interactions, a solution was required which fulfilled three boundary conditions. Firstly the main interaction phenomena should be represented. The outcome of this is that the physics are well enough represented to obtain reliable results which represent real jettisoning simulations without any requirements for helicopter or external body dependent tuning factors. Secondly the required computational simulation wall time for a whole envelope of some hours should remain in the same order of magnitude. This excluded solutions like unsteady CFD-simulations where all interaction phenomena would have been captured very well, but would require months to calculate a whole jettisoning envelope. Thirdly the development time of the interaction method and its integration in the existing simulation process was strongly limited due to limited resources and project deadlines. Therefore it was just possible to choose a solution which could be integrated in existing tools with existing interfaces.

Based on the three boundary conditions a method called difference vector method was developed. The approach of the method is to correct the inflow vector of the external jettisoning body by the influence of the fuselage of the helicopter. The approach takes advantage of two facts – the flow redirection influence of the helicopter on the flow of the external is much bigger than the other way around and the jettisoning process is an initial value problem with decreasing influence of forces and moments on the jettisoning trajectory over time. This means that the final trajectory is predominantly determined by the initial boundary conditions. Therefore the approach is to determine the initial flow conditions for the external body in its mounting position as good as possible for any flight condition. And since besides the main rotor downwash the inflow vector on the external is predominantly influenced by the helicopters fuselage, this influence needs to be included in the simulation. The solution is to create a database which can provide the right flow vector for the external body at its mounting position for any flight condition. This requirement can be fulfilled by a normalized spherical polar which is based on CFD calculations of the helicopters fuselage called the difference vector polar. The method to build a difference vector polar is similar to the method of creating a force and moment polar of the external. Therefore the polar generation procedure described in the chapter “Process improvements” can be used with slight modifications. The fuselage of the helicopter is first simulated in CFD multiple times with changing inflow vectors of constant absolute values. Then the flow vectors at the installation positions of the externals are extracted from the simulated flow fields. The differences between the inflow vectors and the extracted vectors get interpolated by the PiTwice-Interpolation tool on a constantly spaced α/β-grid and stored in a common α/β-polar. This polar is then used directly in the simulation environment and also this procedure is fully automated. In the simulation process the inflow and main rotor downwash velocity is taken from the trimmed state of the helicopter-simulation for each flight state as inflow vector. With this information the according difference vector is extracted and scaled from the difference vector polar. This difference vector is then added to the original inflow vector and the result is used as the new initial inflow vector of the external body. The difference vector is internally simulated as a gust, and is also faded out when the external body departs from the helicopter. By implementing the influence of the fuselage on the external as a polar based difference vector, all three boundary conditions could be fulfilled. The most important interaction influences are represented, the simulation time just increased marginally and all existing interfaces and procedures could be used without major modification.

4.2 Accuracy of the collision detection simulation

Originally the method to detect collisions between the helicopter and the jettisoned external body was limited to two possibilities. 3D-files including every time-step of the jettisoning process could be generated by the trajectories and 3D-Bodies of the helicopter and the jettisoned external. These 3D-files can be loaded and visually analyzed on collision between the two bodies. However this cannot be done manually for each of the approximately 1300 flight states of a single envelope, therefore an automatic collision detection procedure is required. A simple and quick method was available which was able to detect collision between an arbitrary helicopter model and a cuboid. Therefore for each external body a cuboid needed to be created which approximately represented the geometry of the jettisoned external body. In order to be conservative, the whole volume of the external body needed to be a part of the cuboid volume, so that the total volume and dimensions tested on collision was significantly larger than the real geometry. Recently an in-house code named PyJAMA (Python Jettisoning and Armament Monitoring Application) was developed which allows the collision check between two arbitrary 3D-Bodies moving and rotating along defined trajectories. The code was slightly adapted and integrated in the simulation environment which allows now for exact collision detection between the real body geometries and therefore leads to less conservative jettisoning envelopes. In addition to the pure collision detection also the distance over time can be written out and a safety distance margin can be set for the collision detection.

4.3 Determination of collision energies

For some applications it is not just of interest whether collision takes place or not during jettisoning, but also to determine the potential impact of a collision. In order to be able to assess the impact of such a collision, a new routine was developed and implemented in PyJAMA which determines the collision energy of two bodies in case of a collision. The maximum collision energy between two bodies is set by the assumption of a completely inelastic collision. In this case the whole relative rotational and translational kinetic energy perpendicular to the collision surface is eliminated by plastic deformation. Since the helicopter is orders of magnitude bigger and heavier compared to the external bodies, it can be assumed, that the external body has after the collision in the worst case the same translational velocity like the helicopters collision point and no relative rotational speed any more. Thus the total energy can be calculated by $E_{colp} = \frac{1}{2} m_{Ext} V_{rel}^2 + \frac{1}{2} I_{Ext} \Omega_{rel}^2$. However it is not possible to differ between the rotational and the translational relative velocity by the approach velocity $\vec{v}_{rel\_colpnt}$. Therefore first the translational velocity V_{rel} is calculated by the velocity difference of the two center of gravities provided in the trajectories by $V_{rel} = |\vec{v}_{rel}|_2 = |\vec{v}_{Ext} - \vec{v}_{HC}|_2$. The difference between the approach velocity $\vec{v}_{rel\_colpnt}$ and the relative translational velocity $\vec{v}_{rel}$ yields the rotational speed of the jettisoned external body. With this information the relative rotational speed Ω_{rel} can be calculated by $\Omega_{rel} = |\vec{\omega}_{Ext}|_2 = |\vec{v}_{rotExt} \cdot \vec{r}_{cg-colp}^{\,-1}|_2$. $\vec{r}_{cg-colp}$ is defined as the distance between the center of gravity and the collision point of the external body. In these equations it is assumed that the rotational speed of the helicopter is zero. This assumption is valid for the jettisoning simulations because the envelope just contains steady, trimmed flight states. With these equations a conservative value for the collision energy is calculated for every jettisoning flight state with collision and the potential damage dependent on the shape of the colliding parts can be assessed.

5 VALIDATION EXAMPLE

In order to proof the functionality of the improved process and physical modeling several validation cases of different external bodies and helicopters were simulated and compared with video recordings of real jettisoning test cases. As an example, a comparison between the simulated and the real jettisoning of an empty Alpha Jet Tank from a NH90 is provided. In order to allow for a direct comparison, the point of view in the simulation visualization was set to the position of the real camera during the flight test. Photos of the real jettisoning at different points in time after the release of the tank were combined with the simulation results at the same points in time in Figure 2-6. The grey-scaled parts are the jettisoning photos of the real helicopter, with the helicopter and the external body in dark grey. The green shaded wireframe, which is laid over the helicopter and the external body, are the simulation results. As long as the green shaded wireframe of the external body simulation is congruent with the dark grey external body in the photo, the simulation matches reality. The flight case is a descent flight at medium forward velocity and a high descent rate. It can be seen that the simulation matches the real jettisoning test very well in all figures. Especially the nose point, which is in this simulation the most critical part because it passes the helicopter very closely is very well represented with a maximum deviation of about 0.1m at 1.0s, depicted in Figure **1**.

The rear part is displaced by about 0.5m at this stage, but the tank center is already about 4.0m below the helicopter so that the deviation at this stage is not relevant any more for the risk analysis. It is more important that in the early points in time of the jettisoning, the drop velocity and the turn rates in all directions are represented well, in order to realize reliable collision detection in critical flight states.

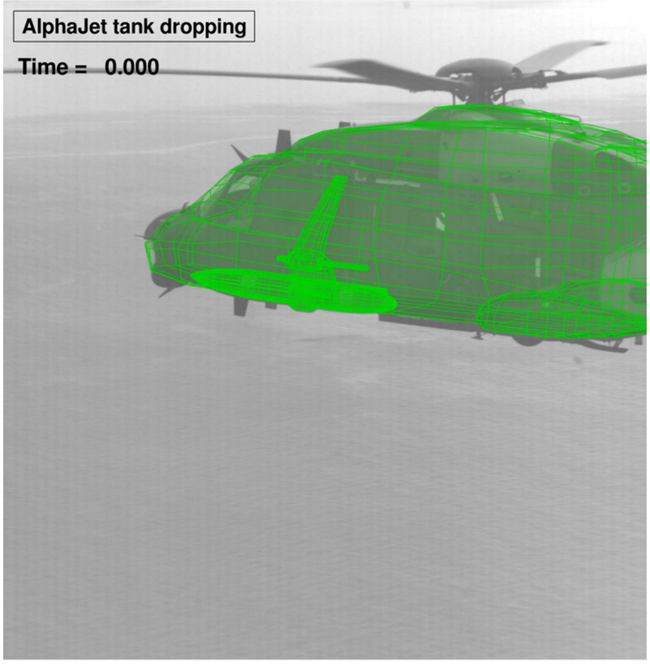


Figure 2: Alpha Jet tank dropping test – *t* =0.00s

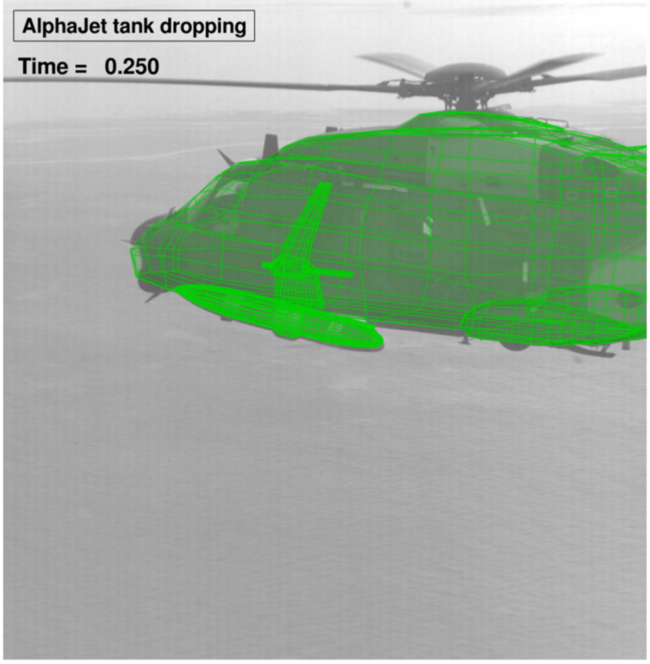


Figure 3: Alpha Jet tank dropping test – *t*=0.25s

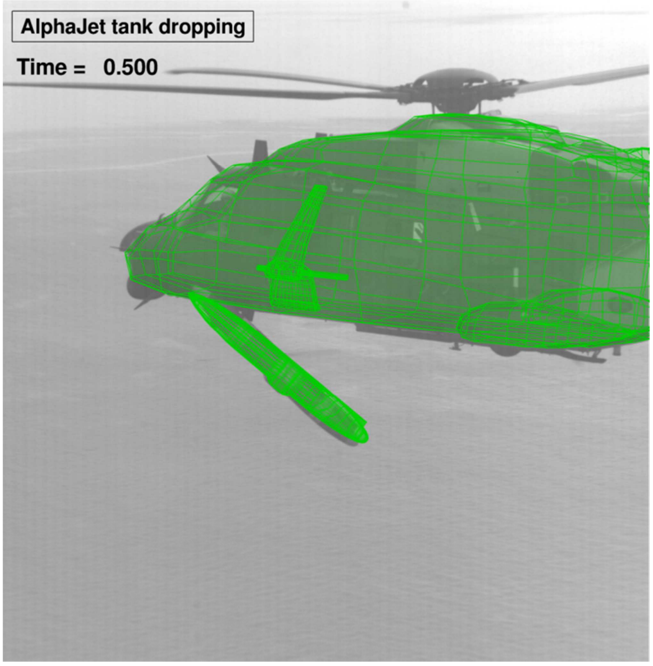


Figure 4: Alpha Jet tank dropping test – *t* =0.50s

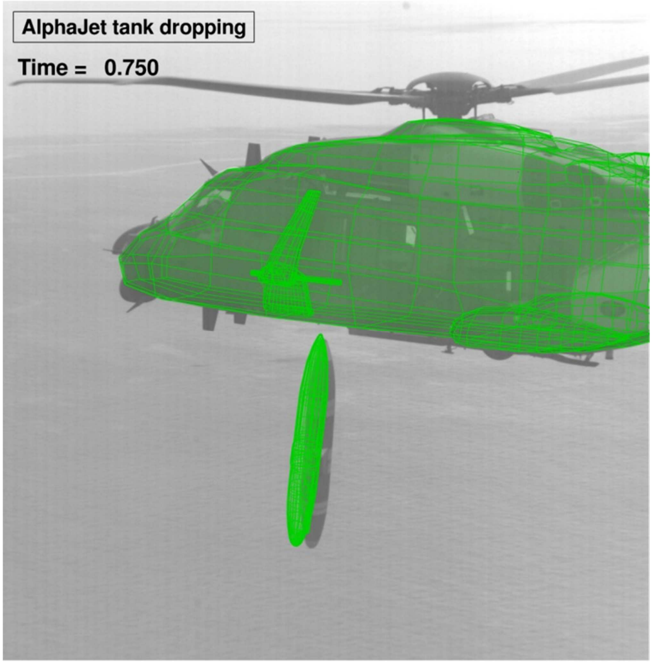


Figure 5: Alpha Jet tank dropping test – *t* =0.75s

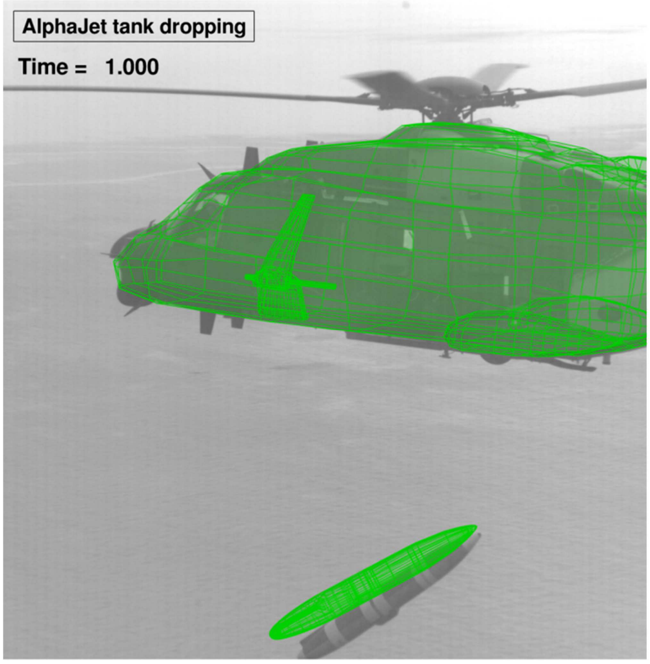


Figure 6: Alpha Jet tank dropping test – *t* =1.00s

6 CONCLUSIONS

In the paper the recent improvements in jettisoning simulation at EC are presented. This comprises improvements in the process chain, especially for the automatic and systematic generation of spherical aerodynamic force- and moment-polars in preprocessing and a significantly simplified setup procedure. A new calculation series management system was implemented in the simulation part and for collision detection, a new exact collision detection method was introduced which was also parallelized to allow for quick simulation results within some minutes to few hours. It also comprises improvements in the physical modeling where a new interaction method was introduced, which represents the influence of the helicopters fuselage on the inflow direction of the jettisoned external. Also the collision detection method now allows for collision detection of the exact geometries of the jettisoned external and is able to calculate the collision energy in case of a contact between the helicopter and the external body. So in summary it can be said, that the existing jettisoning simulation environment at EC was significantly improved in terms of process automation, reproducibility, error-proneness, physical representation and predictive capability.

ACKNOWLEGEMENTS

The authors would like to thank their colleagues Frédéric Guntzer for the development and programming of the collision detection program PyJAMA, Anthony Binder for the development work and programming of the interpolation program $\pi \times II$ (Pi-Twice) and Andreas Kellerer for the establishment of the initial jettisoning process.

COPYRIGHT STATEMENT