The 5th Joint International Conference on Multibody System Dynamics
June 24 – 28, 2018, Lisboa, Portugal

Bifurcation analysis of landing gear shimmy using flexible multibody models

Camiel J. J. Beckers[1], A. Emre Öngüt[2], Bert Verbeek[3], Rob H. B. Fey[1], Yves Lemmens[2] and Nathan van de Wouw[1,4,5]

[1]*Dept. of Mechanical Engineering, Eindhoven University of Technology, The Netherlands,{c.j.j.beckers, r.h.b.fey, N.v.d.Wouw}@tue.nl*
[2]*Siemens PLM Software, {emre.ongut, yves.lemmens}@siemens.com*
[3]*Fokker Landing Gear B.V., bert.verbeek@fokker.com*
[4]*Civil, Environmental & Geo- Engineering Dept., University of Minnesota, U.S.A.*
[5]*Delft Center for Systems and Control, Delft University of Technology, The Netherlands.*

ABSTRACT — *Shimmy oscillations are undesired vibrations in aircraft landing gears. The onset of shimmy, marked by Hopf bifurcations, is investigated in the parameter space of an industrial, high-fidelity, flexible multibody landing gear model. The bifurcation analyses are performed by coupling the Virtual.Lab Motion multibody solver with the numerical continuation software AUTO. The resulting quasi-2-parameter bifurcation diagrams, involving aircraft velocity and vertical landing gear load, are verified using conventional time-domain simulations and are shown to guarantee significantly higher solution accuracy at a similar computational cost.*

1 Introduction

Shimmy is a type of dynamic landing gear instability that can occur during take-off, landing, or taxiing of an aircraft. In severe forms, it can damage the safety-critical landing gear structure, thus increasing the maintenance costs and, in some cases, even leading to catastrophic situations. Therefore, assessment of the susceptibility of landing gears to shimmy early in the design process is of substantial importance. The conventional strategy during the landing gear design is to perform time-domain simulations to investigate whether shimmy occurs during various landing, take-off, and taxi scenarios. This procedure is computationally intensive, and hence time consuming, and does not yield the complete shimmy-free operational parameter range but only ensures that shimmy does not occur on the preselected operational conditions.

The current models used by the landing gear industry can be highly complex and can contain different types of nonlinearities and dynamic effects, such as mechanical free-play, flexible bodies with large rotations and deformations, nonlinear springs and dampers, contacts, and nonlinear tire models. Such multibody models form an accurate representation of the real landing gear dynamics. However, the time-domain simulations involving these models are computationally expensive and can only result in stability properties of the landing gear at discrete points in the parameter space. Numerical continuation methods offer an accurate alternative for this shimmy analysis procedure, where these methods are employed to analyze the Hopf bifurcations that characterize the onset of shimmy. However, currently known applications are limited to relatively small analytical models with significant simplifications with respect to the complex landing gear structures, such as in [1] and [2].

This study brings together numerical continuation methods and multibody analysis software to determine the shimmy stability of a complex, high-fidelity, and flexible multibody model. The simulation framework developed in [3] is adapted and improved to enable the analysis of these type of larger and more complex models. In contrast with [3], the current model represents a real-world landing gear with more bodies, some of which are considered flexible to account for the compliance of the landing gear structure.

2 Advanced Multibody Dynamics

In general, the dynamics of a rigid multibody system can be described by a system of Differential-Algebraic Equations (DAE's) according to [4, p. 244]

$$\begin{cases} \begin{bmatrix} \underline{M}(\underline{q}) & \underline{\Phi}_{,\underline{q}}^T(\underline{q},t) \\ \underline{\Phi}_{,\underline{q}}(\underline{q},t) & 0 \end{bmatrix} \begin{bmatrix} \ddot{\underline{q}} \\ \underline{\lambda} \end{bmatrix} &= \begin{bmatrix} \underline{Q}^A(\underline{q},\dot{\underline{q}}) + \underline{Q}^E(\underline{q},\dot{\underline{q}},t) \\ \underline{\gamma}(\dot{\underline{q}},\underline{q},t) \end{bmatrix}, \\ \underline{\Phi}_{,\underline{q}}(\underline{q},t)\dot{\underline{q}} &= -\frac{\partial \underline{\Phi}(\underline{q},t)}{\partial t}, \\ \underline{\Phi}(\underline{q},t) &= \underline{0}. \end{cases} \tag{1}$$

In this system of equations, $\underline{q}$ is the column of generalized coordinates, containing both the displacements $\underline{r}$ and the rotations, in the form of Euler parameters $\underline{\theta}$, that describe the position and orientation of the bodies in the multibody system. Furthermore, $\underline{M}$ is the mass matrix, $\underline{\Phi}_{,\underline{q}}$ is the constraint Jacobian matrix, where $_{,\underline{q}}$ denotes the partial derivative with respect to $\underline{q}$ of the holonomic constraints $\underline{\Phi} = \underline{0}$, $\underline{\lambda}$ is the associated column of Lagrange multipliers, and $\underline{Q}^A$ denotes the column of generalized applied forces. Additionally, in the Virtual.Lab Motion software, the column $\underline{Q}^E$ includes applied forces supplied by external sources, such as an external control system. The column $\underline{\gamma}$ defines the acceleration-level constraint equations, according to

$$\underline{\Phi}_{,\underline{q}}(\underline{q},t)\ddot{\underline{q}} = -\left[\frac{\partial\left(\underline{\Phi}_{,\underline{q}}(\underline{q},t)\dot{\underline{q}}\right)}{\partial \underline{q}} + 2\frac{\partial \underline{\Phi}_{,\underline{q}}(\underline{q},t)}{\partial t}\right]\dot{\underline{q}} - \frac{\partial^2 \underline{\Phi}(\underline{q},t)}{\partial t^2} =: \underline{\gamma}(\dot{\underline{q}},\underline{q},t), \tag{2}$$

while the second and third equalities in (1) represent the constraint equations on velocity- and position-level, respectively.

The Motion solver, included in the Virtual.Lab Motion software package, is used in this study. In this solver, the second-order differential equations in (1) are transformed into a first-order form by introducing the state variable $\underline{x} = [\underline{x}_1^T\ \underline{x}_2^T]^T := [\underline{q}^T\ \dot{\underline{q}}^T]^T$, to support further analysis. The dynamics in (1) can now be described in terms of the variables $\underline{x}$, $\underline{\lambda}$, and $\underline{\chi}$ as

$$\begin{bmatrix} \underline{I} & 0 & 0 & 0 \\ 0 & \underline{M}(\underline{x}_1) & \underline{\Phi}_{,\underline{x}_1}^T(\underline{x}_1,t) & 0 \\ 0 & \underline{\Phi}_{,\underline{x}_1}(\underline{x}_1,t) & 0 & 0 \\ 0 & 0 & 0 & \underline{I} \end{bmatrix} \begin{bmatrix} \dot{\underline{x}}_1 \\ \dot{\underline{x}}_2 \\ \underline{\lambda} \\ \dot{\underline{\chi}} \end{bmatrix} = \begin{bmatrix} \underline{x}_2 \\ \underline{Q}^A(\underline{x}_1,\underline{x}_2) + \underline{Q}^E \\ \underline{\gamma}(\underline{x}_1,\underline{x}_2,t) \\ \underline{g}(\underline{x}_1,\underline{x}_2,\underline{\lambda},\underline{\chi}) \end{bmatrix}, \tag{3}$$

which, in case of time-simulation, are integrated directly using one of the available numerical integration schemes. The additional states $\underline{\chi}$ in (3) represent the dynamics of the system that are not directly a part of the multibody dynamics, for instance related to the tire dynamics. Note that the augmented formulation is used and that $\underline{x}$ thus contains states corresponding to both dependent and independent generalized coordinates. States corresponding to the latter are shown with $\underline{x}_I$.

2.1 Including Flexible Bodies

The dynamics of a rigid multibody model, as presented in (3), can be extended by taking the structural flexibility of one or more bodies into account. A full description of the theory that is used to derive the equations of motion of such a flexible multibody system is provided in [5] and [6], and more generally in [7]. A brief summary of the applied method is presented below.

In order to include the structural compliance of a body B in the equations of motion of the multibody system, first, a modal representation of the elastic deformation of the body is required. To this end, a lumped-mass finite element mesh of the body is created and Craig-Bampton modal reduction [8] is applied to describe the local

deformation of the body according to $\underline{u} = \underline{\Psi}\underline{\eta}$, where $\underline{u}$ is the $(n_u \times 1)$ column containing the n_u displacement degrees of freedom (DOFs) of the nodes of the finite element mesh, $\underline{\Psi}$ is the $(n_u \times n_\eta)$ matrix containing the modes resulting from the Craig-Bampton modal reduction, and $\underline{\eta}$ is the $(n_\eta \times 1)$ column containing the n_η modal coordinates. As the Craig-Bampton method is used to determine $\underline{\Psi}$, the modes in the columns of this matrix are a combination of static constraint modes and fixed interface normal modes. The static constraint modes represent deformations due to unit displacements of the boundary DOFs, whereas the fixed interface normal modes are the eigenmodes of the bodies with all boundary DOFs fixed. In this analysis, DOFs of the mesh connected to the articulation elements in the multibody simulation, such as joints or drivers, are considered to be boundary DOFs, whereas all remaining DOFs are internal DOFs.

To include this description of the elastic deformation of a body in the equations of motion of the multibody system, the Motion solver uses a floating reference formulation. This implies that the displacements due to the elastic deformation of each flexible body are described with respect to a body-fixed reference frame. This procedure is explained in detail in [6]. The result is that the configuration of each flexible body is not merely a function of the reference coordinates $\underline{r}$ and $\underline{\theta}$, that respectively describe the position and orientation of the body-fixed reference frame, but also of the modal coordinates $\underline{\eta}$, that describe the elastic deformation with respect to the body-fixed reference frame. Therefore the global position of any node on the flexible body is now fully defined by the generalized coordinates $\underline{q} = \left[\underline{r}^T \ \underline{\theta}^T \ \underline{\eta}^T\right]^T$. This expression shows that including structural compliance in the simulation includes the modal coordinates directly in the set of generalized coordinates, which can significantly increase the number of states $\underline{x}$ in (3).

2.2 Tire Dynamics

To simulate the tire dynamics, a Virtual.Lab Motion featured tire model is used. This model, named the "Complex Tire" model [9], calculates the forces between the road and the tire, as a function of the tire's position and velocity relative to the road. These forces are then included in the generalized applied forces $\underline{Q}^A$ in (3).

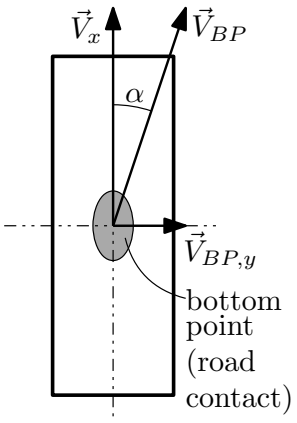


Fig. 1: Top view of a tire with side slip angle definition

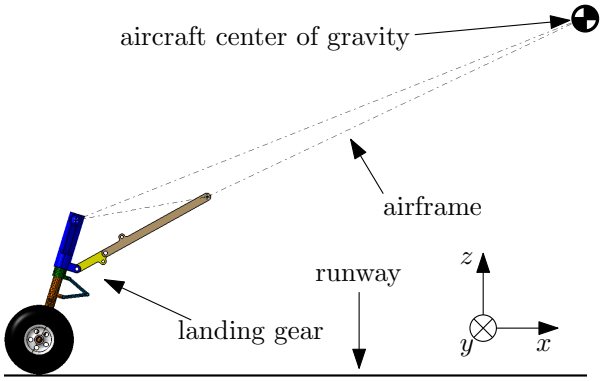
Of these forces, the lateral tire force F_y and the self-aligning moment M_z are most important for properly describing shimmy behaviour in single-wheeled landing gears, because the shimmy oscillations are characterized mostly by lateral and yaw motion of the tire. In the particular tire model that is used, the instantaneous lateral force F_y is a non-linear function of the sideslip angle α. This angle is defined as the acute angle between the longitudinal tire velocity direction, $\vec{V}_x$, and the velocity vector of the bottom point of the tire, $\vec{V}_{BP}$, that is in contact with the road. These vectors are visualized in Fig. 1. For situations where the side slip angle is relatively small, the instantaneous lateral force F_y is characterized by the cornering stiffness $C_{f\alpha}$ according to $F_y = C_{f\alpha}\alpha$ for $\alpha \ll 1$, where $C_{f\alpha}$ is a user supplied tire parameter, and is considered to be constant. The instantaneous self-aligning moment is defined by $M_z = -r_p F_y$, where r_p is the pneumatic trail, which is also assumed to be constant.

Due to relaxation of the pneumatic tire, the response of the lateral force to side slip is not instantaneous. In this particular tire model, this phase lag is modeled by introducing a relaxation time t_σ for the lateral force and self-aligning moment, according to

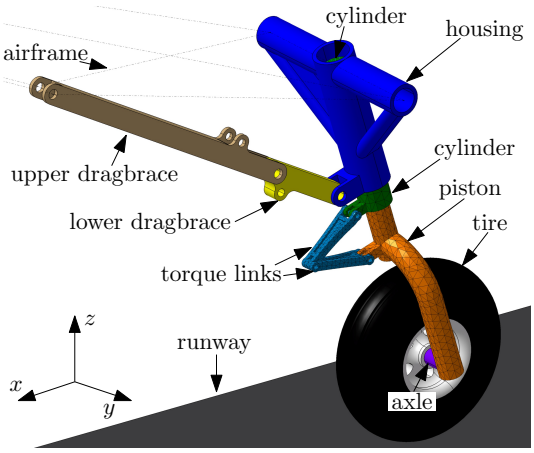
$$\dot{\tilde{F}}_y = \frac{1}{t_\sigma}\left(F_y - \tilde{F}_y\right), \quad \dot{\tilde{M}}_z = \frac{1}{t_\sigma}\left(M_z - \tilde{M}_z\right), \tag{4}$$

where the relaxation time is derived from the relaxation length σ according to $t_\sigma = \sigma/|\vec{V}_x|$, where σ is considered to be a constant, user supplied tire parameter. In (4), $\tilde{F}_y$ and $\tilde{M}_z$ are respectively the resulting lateral force and self-aligning moment that include the relaxation behavior. These, together with all other calculated tire forces, are included in $\underline{Q}^A$, after taking the friction ellipse limit into account.

For every tire present in the multibody model, the two differential equations in (4) are added to the system of DAE that are analyzed by the Motion solver. Therefore, $\tilde{F}_y$ and $\tilde{M}_z$ appear as additional states $\underline{\chi}$ in (3).



(a) Side view of the study model.



(b) Isometric view of the study model, including the names of the separate bodies.

Fig. 2: Side and isometric views of the landing gear model that is investigated in this study.

3 Multibody Model of a Nose Landing Gear

In order to conduct a relevant shimmy study, an industrial landing gear model, which is referred as the 'industry model' here, is provided by Fokker Landing Gear. This model is replicated using simplified geometry but with similar modeling elements. This replicate is referred to as the 'study model', as it will be used for the bifurcation study.

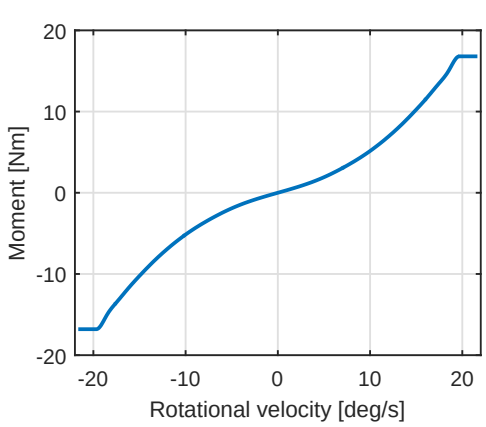
A side view of the study model is shown in Fig. 2a. The figure shows the landing gear structure, the runway, and the airframe. The airframe, which is illustrated with dotted lines, is a rigid frame representing the fuselage of the aircraft to which the landing gear is attached. It has a mass that induces the nominal vertical force on the landing gear structure. The airframe is constrained to move only vertically with respect to the fixed global axis system, so the landing gear can drop down onto the runway. The runway is moved backwards relative to the landing gear to simulate a forward motion of the aircraft. A more detailed, isometric view of the landing gear is illustrated in Fig. 2b. The housing, upper drag brace, and the lower drag brace are represented as rigid bodies and these are constrained such these they cannot move with respect to the airframe or to each other. Therefore, the airframe, upper- and lower drag brace, and the housing move as one rigid body.

The flexible cylinder body is fitted inside the housing. Between these two bodies, only rotation about their co-axial axis is allowed. During taxiing, this DOF enables the steering of the nose landing gear with a steering actuator. During landing or take off, the steering actuator is turned off, where it acts as a hydraulic shimmy damper. A quadratic damping curve is used to model this hydraulic damping, which is shown in Fig. 3a.

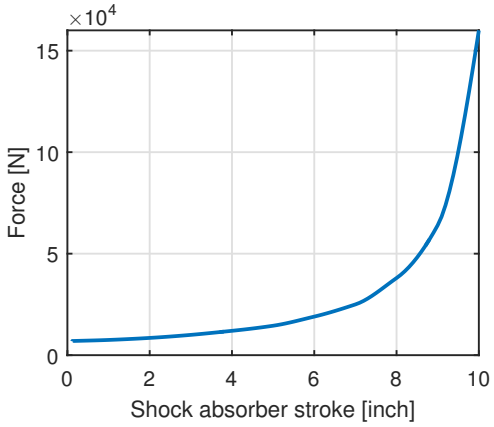
The piston can move axially with respect to the cylinder and rotate about their co-axial axis as well. In reality, this motion is enabled by two bearings that guide the sliding and rotating motion of the piston with respect to the cylinder. In the multibody model, flexible-point-curve-joint constraints are used to simulate these bearings and constrain the two flexible bodies together. The shock absorber force acts between the piston and the cylinder along their common axis. The static shock absorber force is modeled by a non-linear, progressive spring, which is shown as a function of shock absorber stroke in Fig. 3b.

The torque links are two bars that are hinged together, that restrict the rotation of the piston relative to the cylinder, without constraining the shock absorber stroke. The torque links determine the yaw-stiffness of the landing gear structure to a large extent. The design of the two torque links is identical. The cylinder, piston, and the two torque links are represented as flexible bodies in the study model.

A rigid axle connects the tire to the piston. The tire is simulated by a rigid body, which is only used to characterize the tire mass and rotational inertia. The forces exerted on the tire by the runway are calculated using the tire model described in Section 2.2.



(a) Nonlinear damping moment curve of the shimmy damper.



(b) Nonlinear shock absorber force as a function of the shock absorber stroke.

Fig. 3: Nonlinear behavior of the shimmy damper and the shock absorber.

3.1 Simplifications Applied to the Industry Model

The industry model that is acquired from Fokker Landing Gear includes some modeling elements that are not copied to the study model. These elements are expected to have only a small influence on the shimmy. There are mainly three simplifications. Firstly, the airframe is treated as a rigid body in the study model while it is flexible in the industry model. Secondly, the free-play is removed from the industry model. This can, for instance, be lateral free-play between the landing gear and the airframe, as well as the free-play in the torque links. These types of free-play can result in small amplitude limit cycle oscillations of the landing gear at all forward velocities. Even though it is possible to determine these oscillations with AUTO, these are left out as they are too small to be considered problematic shimmy and periodic solutions are computationally more challenging to follow in a continuation analysis. The third effect that is not taken into account in the study model is the rotational friction of the bearings between the piston and the cylinder. In a shimmy analysis, friction tends to hide shimmy behavior, as it dissipates energy from the system. Therefore, neglecting friction can be considered as a worst-case assumption in the context of shimmy analysis.

These simplifications imply that the dynamic behavior of the study model will be somewhat different from the industry model. However, the main effects that influence the shimmy behavior, such as the tire dynamics and viscous shimmy damper properties, remain unchanged.

3.2 Model Dimension

There are four flexible bodies in the study model; the cylinder, the piston, and the two torque links. Craig-Bampton mode sets are used to represent the deformation of the flexible bodies. To this end, a finite element mesh is created for each body, and the interface DOFs are determined, based on the joints and drivers attached to each body. Static constraint modes are determined for each body according to the theory in Section 2.1.

Due to the various joints and force-elements connected to the flexible bodies, 53 static constraint modes are required to represent the static stiffness of the four flexible bodies: 10 for each torque link, 18 for the piston, and 15 for the cylinder. In order to reduce the model dimension sufficiently, no fixed interface normal modes are included in the mode sets for the flexible bodies. Therefore, the total number of modal coordinates is given by the number of static constraint modes minus the 4×6 rigid body modes. This results in 29 modal coordinates.

The reason why the normal modes are omitted is that it was computationally challenging to run a bifurcation analysis with such a large number of modes. In order to ensure that the accuracy of this approach is acceptable, the model is compared to an "unreduced model" which contained 10 fixed interface normal modes in addition to the static constraint modes for each flexible body. The results showed that especially the first lateral bending and torsional modes matches the unreduced model quite well. A relatively larger difference is observed in the

eigenfrequency of the first longitudinal bending mode, of roughly 8 %. However, for a single-wheeled landing gear, such as the one presented here, the longitudinal structural dynamics have relatively small effect on the shimmy. Therefore, it is concluded that the modal reduction strategy is accurate enough for shimmy analysis.

The resulting multibody study model contains 16 bodies and 146 generalized coordinates in total. Together with the 16 Euler parameter constraints, the joints and drivers in the model introduce a total of 114 constraint equations. By subtracting this number from the total number of generalized coordinates, the total number of independent generalized coordinates can be determined, which is 32. Therefore, the total first-order representation of the equations of motion, as described by (3), contains 66 independent states $\underline{x}_I$, as there are also two tire states $\underline{\chi}$ present.

4 Continuation Analysis of a Multibody System

The recently developed simulation framework by Tartaruga et al. [3] is adapted to perform bifurcation analyses on the study model as described in Section 3. The continuation software AUTO [10] is coupled to the Virtual.Lab Motion multibody solver via MATLAB to realize the bifurcation analysis on a multibody model. A custom version of the Motion solver that includes a MATLAB interface is used. This interface makes all states $\{\underline{x}, \underline{\lambda}, \underline{\chi}\}$ of the first-order equations of motion (3) and the bifurcation parameters available in the MATLAB-workspace. To be able to use AUTO within MATLAB, the Dynamical Systems Toolbox (DST) is used [11], which is an open source MATLAB toolbox that encapsulates all the functionalities of the bifurcation software AUTO.

4.1 Simulation Framework

The general outline of the bifurcation analysis framework is illustrated in Fig. 4. Given a viable set of independent states $\{\underline{x}_I, \underline{\chi}\}$ and the bifurcation parameter p, the Motion solver is able to algebraically solve the first-order equations of motion (3), which results in the time derivatives of the independent states $\underline{\dot{x}}_I$ and $\underline{\dot{\chi}}$. These are used by the DST, and the underlying AUTO algorithms, to identify the (quasi-)static solutions, the stability properties, and the bifurcation points of the system. This is an iterative process, where the DST constantly updates and applies perturbations to the states of the model. Using this methodology, branches of (quasi)-static solutions can be followed in the parameter space.

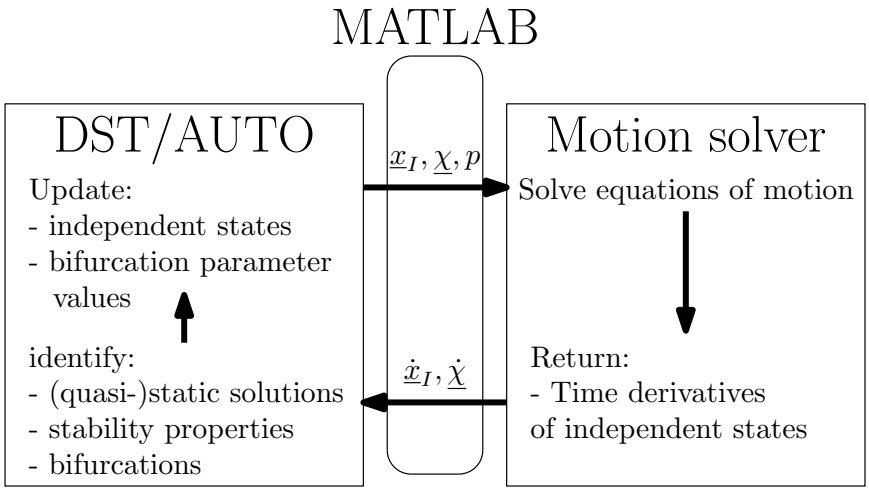


Fig. 4: Schematic of the coupling between the Dynamical Systems Toolbox (DST) and the VLM solver through MATLAB

A more detailed flowchart of the coupling between the Motion solver with the DST/AUTO is presented in Fig. 5, where individual steps are indicated by { }. The procedure is controlled by MATLAB, that is to say, all functionality is accessed via functions in MATLAB. The solid lines represent the flow of the execution, whereas the dashed lines represent the data exchange between MATLAB and Motion solver and the dotted lines represent

the flow of information within Motion solver. At the beginning of the simulation, the Motion-MATLAB interface is loaded with the `vlmLoadLibrary` function $\{1.1\}$. The interface makes various functions available in the MATLAB workspace, which allow the control of the Motion solver. Next, a multibody model is loaded into the memory of Motion solver with `vlmLoad` function by providing a solver input file $\{1.2\}$ as an argument. This input file contains all the information of the model required for the solver to construct the DAE of motion, as described by (3). After the model is loaded, the initial state values $\{1.3\}$ and the solver suggested independent states indices $\{1.4\}$ are retrieved with the `vlmInitialState` and `vlmIndependentIndices` functions, respectively.

After the MATLAB environment is initialized and the multibody model is loaded, the second stage of the solution is performed $\{2.1\text{-}2.4\}$. During this stage, a (quasi-)static initial solution is found by performing a time simulation with the Motion solver. This initial solution is required by AUTO to start the bifurcation analysis. During this initial time integration, the landing gear touchdown on the runway is simulated. The system is integrated in time until the transient dynamics of the landing have damped out and the (quasi)-static solution remains.

The time integration is performed as a co-simulation between MATLAB and Motion solver, which is marked as 'loop 1' in Fig. 5. At every communication step of this loop, the equations of motion are integrated over a small time step Δt by the Motion solver and the updated state values are returned to MATLAB. During this simulation, constant values for the bifurcation parameters are imposed. In case the bifurcation parameter is a force, its value is imposed on the equations of motion through the control-interface of Virtual.Lab Motion, $\underline{Q}^E$ in (3), during the integration process. However, if the bifurcation parameter is a state in the equations of motion, such as the velocity of a body, the state is directly overwritten in the column of independent states $\underline{x}_I$. Note that this approach is theoretically valid only if the state on which a value is imposed is decoupled from other DOFs in the model. In order to make sure that this is true for the current application, the velocity constraint is applied to the runway body, which is unconstrained from other bodies in the forward direction. This strategy ensures the theoretical validity of overwriting the state value for the bifurcation parameter. In summary, at every communication step, the value of the body velocity is imposed within the set of independent states $\underline{x}_I$, after which the dependent states are calculated with `vlmDependentStates` function $\{2.2\}$. The resulting set of states is used as the starting point for the integration step with the `vlmStep` function $\{2.3\}$ and the loop is repeated. When the first time derivatives of the independent states of the first-order equations of motion are below a user-defined tolerance, the solution is considered to be close enough to the final (quasi-)static equilibrium solution, and the values of all states at the final time step are stored to be used in the bifurcation analysis $\{2.4\}$.

At the third step of the simulation framework, the continuation analysis is performed on the multibody system. During the continuation analysis, a branch of (quasi-)static solutions is followed in the bifurcation parameters space. For every solution, the stability properties are determined by analyzing the eigenvalues of the Jacobian of the linearized systems equations. These eigenvalues are also used to detect possible Hopf bifurcations, which are defined by a complex conjugate pair of eigenvalues crossing the imaginary axis transversely [12]. By monitoring the number of eigenvalues of the Jacobian in the left-half complex plane together with the real part of the eigenvalue closest to the imaginary axis [13], the AUTO software can detect Hopf bifurcations.

The continuation algorithm is indicated with 'loop 2' in Fig. 5. The first two function calls $\{3.2\}$ and $\{3.3\}$ are the same as the ones in $\{2.2\}$ and $\{2.3\}$ and serve the purpose of algebraically solving the equations of motion as a function of the independent states and the bifurcation parameter. Instead of performing a time integration, the `vlmStep` function is called with a time step of 0, which leads the Motion solver to solve the equations of motion algebraically with $\{\underline{\dot{x}}_I, \underline{\lambda}, \underline{\dot{\chi}}\}$ as the unknowns. After solving the equations of motion, possible control outputs $\{3.4\}$ and time derivatives of the states $\{3.5\}$ are returned to MATLAB with `vlmOutputs` and `vlmDerivatives` functions, respectively. Using this strategy, the tools available in DST/AUTO, such as the continuation of static solutions and detection of bifurcations, can be applied to analyze the multibody system.

Once the continuation algorithm has completed all computations, the results are plotted for user evaluation $\{4.1\}$. Finally, the model is closed and unloaded from the Motion solver $\{4.2\}$ before unloading the MATLAB-Motion solver interface $\{4.3\}$.

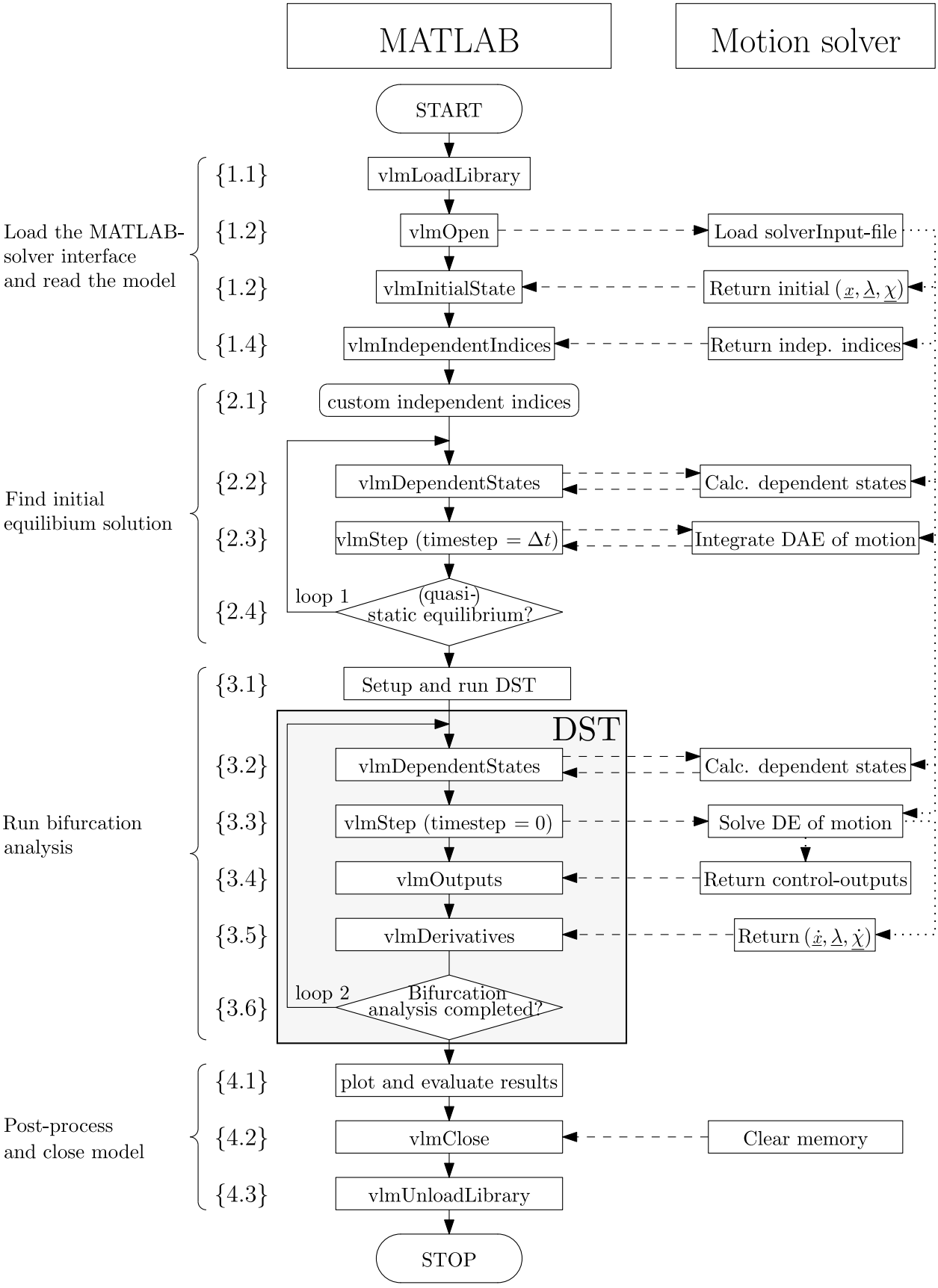


Fig. 5: Flowchart of the entire bifurcation analysis, highlighting the communication between MATLAB and the Motion solver through use of the custom functions supplied by the interface.

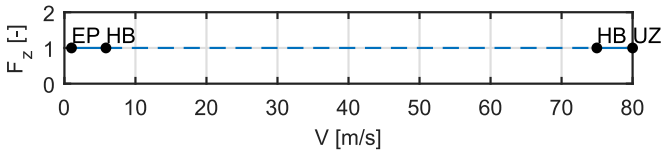


Fig. 6: 1-parameter bifurcation diagram. 'EP' and 'UZ' indicate the start and end of the solution branch respectively. HB indicates a supercritical Hopf bifurcation.

5 Landing Gear Shimmy Analysis Using Bifurcation Methods

The simulation framework presented in Section 4 is used to analyze the possible occurrence of shimmy in the parameter space of the flexible landing gear multibody study model presented in Section 3. In the current study, the relevant (and safety-critical) case of a leaking shimmy damper is researched. To emulate the loss of damping in the shimmy damper, the damping moment curve shown in Fig. 3a is reduced by a factor of 0.03.

5.1 1-Parameter Bifurcation Analysis

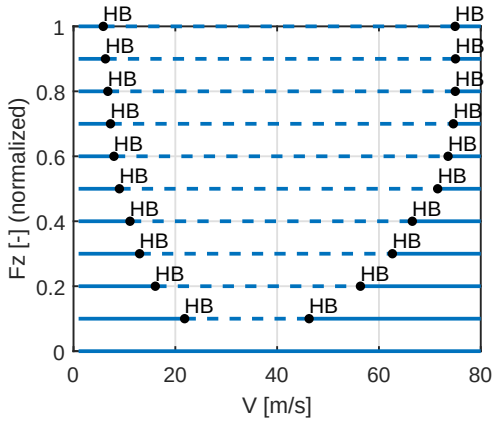
A 1-parameter bifurcation analysis is performed by continuing a branch of (quasi-)static solutions for increasing values of runway velocity V, while the vertical force F_z remains constant. The relative velocity of the runway with respect to the landing gear V is chosen to be the primary bifurcation parameter as the velocity has a large influence on the occurrence of shimmy and it can vary significantly during landing and take-off. The result of this 1-parameter bifurcation analysis is shown in Fig. 6. The solid lines indicate stable (quasi-)static solutions, whereas the dashed line indicates unstable (quasi-)static solutions. The vertical force is normalized at 20 kN, while the velocity changes from 1 m/s to 80 m/s. The (quasi-)static starting solution is determined using time simulation, as described in step $\{2\}$ of Fig. 5, and is marked by 'EP'. The end point of the solution branch is marked by 'UZ'. This is the maximum landing/take-off velocity for this type of landing gear. Along the solution branch, two super-critical Hopf bifurcations are encountered at 5.8 m/s and 75 m/s respectively, both marked by 'HB'. These indicate the edges of a domain where the stationary (quasi-)static solution becomes unstable and a coexisting stable oscillatory solution emerges. Therefore, the range indicated by the dashed line is the velocity range for which the landing gear model shows shimmy.

5.2 Quasi-2-Parameter Bifurcation Analysis

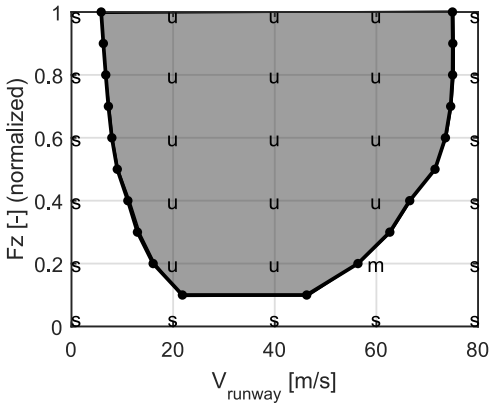
The 1-parameter bifurcation analysis can be repeated for different values of the normal force on the landing gear F_z. By doing so, the stability of the landing gear is evaluated in the two parameter space (V, F_z). The results of this quasi-2-parameter bifurcation study are visible in Fig. 7a. Individual 1-parameter bifurcation analyses show the stability of the equilibrium solutions at constant values for F_z. By connecting the Hopf bifurcation points for multiple values of F_z, the stability boundary in the 2-parameter plane can be visualized, as indicated by the black line in Fig. 7b. In this figure, the unstable parameter domain is shaded. The results in Fig. 7b show that for most vertical loads, this landing gear is stable at low velocities, then becomes unstable for increasing velocities and then regains its stability again at the highest velocities.

The results of the bifurcation analysis are verified through a comparison with standard time simulations of the same multibody model. An aircraft landing is simulated by gradually lowering the landing gear on the moving runway under the influence of gravity and the applied vertical force F_z. After the transient behavior has damped out, the stability properties of the steady-state solution of the model are assessed. This process is repeated for various parameter value combinations of V and F_z. The results are shown in Fig. 7b with letters 's', 'u', and 'm', indicating a stable (quasi-)static, unstable, and marginally stable solution, respectively. The comparison of time simulations and bifurcation analysis reveals that the two methods are in accordance.

For the results shown here, both methods require a similar amount of computation time. However, the grid of solutions in the (V, F_z)-parameter space found with the bifurcation analysis is much denser and even reveals the



(a) Results of multiple 1-parameter bifurcation analyses, where 'HB' indicates the supercritical Hopf bifurcations. Solid lines indicate locally stable solutions, whereas dashed lines mark local instability of the (quasi-)static solution.



(b) The same stability boundary as of Fig. 7a is indicated by a black line and the results of 30 time-domain simulations are indicated by markers: 's' is stable, 'u' is unstable, and 'm' is marginally stable. The unstable domain is shaded.

Fig. 7: Results of the quasi-2-parameter bifurcation analysis.

exact location of 20 points on the stability boundary. Achieving the same kind of accuracy with the conventional time-simulation method would require more simulations and a significantly higher computational burden.

6 Conclusions

In this study, the shimmy analysis of a complex, high-fidelity, flexible multibody model of a landing gear is performed using continuation algorithms. The recently developed simulation framework that combines the continuation software DST/AUTO with Virtual.Lab Motion solver is adapted to perform the bifurcation analyses of the landing gear multibody model. The bifurcation analysis successfully detected the stability boundary of the landing gear in the (V, F_z)-parameter space.

The scenario where the shimmy damper has a leakage is investigated by lowering the damping value. The analyses are performed in the form of 10 1-parameter bifurcation analyses at different F_z values with V as the bifurcation parameter. In total, 20 Hopf bifurcations are detected, which yields the unstable domain in a discrete yet accurate way.

The computational cost of the bifurcation analysis is found to be in the same order as the shimmy analysis by running time-domain simulations of the multibody landing gear model in Virtual.Lab Motion at discrete points in the parameter space. Therefore, in terms if computation time the presented method performs same as conventional simulation strategies. However, the bifurcation analysis gives a more complete and accurate representation of the landing gear instabilities compared to the trial and error approach that has to be employed during time-domain simulations. With the developed methodology, the bifurcation analysis can serve as a welcome addition to the current methods for detecting and simulating shimmy in high-fidelity, complex landing gear multibody models.

References

[1] C. Howcroft, M. H. Lowenberg, S. Neild, B. Krauskopf, and E. Coetzee, "Shimmy of an aircraft main landing gear with geometric coupling and mechanical freeplay," *J. Comput. Nonlinear Dyn.*, vol. 10, p. 051011, Sept. 2015.

[2] P. Thota, B. Krauskopf, and M. Lowenberg, "Interaction of torsion and lateral bending in aircraft nose landing gear shimmy," *Nonlinear Dyn.*, vol. 57, no. 3, pp. 455–467, 2009.

[3] I. Tartaruga, M. H. Lowenberg, J. E. Cooper, P. Sartor, and Y. Lemmens, "Bifurcation analysis of a nose landing gear system," in *15th Dyn. Spec. Conf.*, (San Diego, California, USA), American Institute of Aeronautics and Astronautics, Jan. 2016.

[4] E. J. Haug, *Computer Aided Kinematics and Dynamics of Mechanical Systems*. 160 Gould Street, Needham Heights, Massachusetts 02194: Allyn and Bacon, 1 ed., 1989.

[5] W. S. Yoo and E. J. Haug, "Dynamics of articulated structures. part I. theory," *J. Struct. Mech.*, vol. 14, pp. 105–126, Jan. 1986.

[6] W. S. Yoo and E. J. Haug, "Dynamics of flexible mechanical systems using vibration and static correction modes," *J. Mech., Trans., and Automation*, vol. 108, no. 3, pp. 315–322, 1986.

[7] A. A. Shabana, *Dynamics of Multibody Systems*. Cambridge University Press, 4th ed., 2013.

[8] M. C. C. Bampton and R. R. Craig Jr., "Coupling of substructures for dynamic analyses.," *AIAA J.*, vol. 6, pp. 1313–1319, July 1968.

[9] Siemens Product Lifecycle Management Software Inc., "LMS Virtual.Lab Motion Online Help (documentation)," 2017.

[10] E. J. Doedel and B. E. Oldeman, "AUTO-07P: Continuation and bifurcation software for ordinary differential equations," tech. rep., Concordia University, Montreal, Jan. 2012.

[11] E. Coetzee, P. Thota, and J. Rankin, "Dynamical Systems Toolbox." `https://seis.bristol.ac.uk/~ec1099/#Dynamical_Systems_Toolbox`, 2015.

[12] J. J. Thomsen, *Vibrations and Stability*. Berlin, Heidelberg: Springer, 2 ed., 2003.

[13] E. Doedel, H. B. Keller, and J. P. Kernevez, "Numerical analysis and control of bifurcation problems (I): Bifurcation in finite dimensions," *Int. J. Bifurc. Chaos*, vol. 01, pp. 493–520, Sept. 1991.