

# GENERIERUNG ANALYTISCHER MEHRKÖRPER-MODELLE MIT UNTERSCHIEDLICHEN MODELLIERUNGSTIEFEN FÜR TILTROTOR VTOL MITTELS DES PRINZIPS NACH JOURDAIN

U. Nolte\*, M. Henkenjohann\*, F. Sion\*, C. Henke\*, A. Trächtler‡

\* Fraunhofer-Institut für Entwurfstechnik Mechatronik IEM,  
Zukunftsmeile 1, 33102, Paderborn, Deutschland

‡ Heinz Nixdorf Institut, Universität Paderborn, Fürstenallee 11, 33102, Paderborn

## Zusammenfassung

In dieser Abhandlung wird ein Vorgehen zur Erstellung von analytischen Mehrkörpermodellen für Tiltrotor VTOL beschrieben. Hierbei liegt der Fokus auf der Ausleitung verschiedener Detaillierungsgrade für diverse Aufgaben des mechatronischen Entwurfs und der einfachen Übertragbarkeit auf unterschiedliche Fluggeräte.

## 1. EINLEITUNG

Im Fraunhofer-Leitprojekt Albacopter® wurde ein Tiltrotor VTOL für den Transport von Gütern zu entwickelt. Die maximale Abflugmasse des Fluggeräts liegt bei 150 kg. Die risikoarme Entwicklung und Inbetriebnahme des Fluggerätes stellen eine wesentliche Rahmenbedingung im Projekt dar. Daher werden im Projekt Fluggeräte unterschiedlicher Skalierung eingesetzt (BILD 1).



BILD 1. Albacopter 0.1 und Albacopter 0.5 auf einem 3DoF-Prüfstand

Neben der modularen Systemarchitektur der Teilsysteme [1], ist ein wesentlicher Bestandteil der mechatronischen Synthese der Tiltrotor VTOL Fluggeräte das modellbasierte Vorgehen zur Auslegung und Erprobung der Regelungstechnik [2] und zur Auslegung der Aktorik. Ein automatisiertes Vorgehen zur Generierung der Bewegungsgleichungen in verschiedenen Modellierungstiefen kann einen wichtigen Beitrag leisten. Die Motivation der verschiedenen Detaillierungsgrade liegt in differenzierten Modellanforderungen unter Berücksichtigung der Berechnungszeit und der physikalischen Detailtiefe der Modelle. Hierbei kann zum Beispiel zwischen Echtzeitanwendungen, Modellen für Optimierungsprozesse und detaillierten Modellen zur Untersuchung spezifischer Phänomene unterschieden werden. Durch die automatische Generierung ergibt sich neben dem zeitlichen Vorteil auch der Vorteil der Konsistenz zwischen

den Detaillierungsebenen. Darüber hinaus kann von einer erhöhten Sicherheit in der Korrektheit des generierten Modells, gegenüber einer klassischen Modellierung, ausgegangen werden, da dieses aus bereits abgesicherten wiederkehrenden Elementen erzeugt wird. Für die Synthese von Reglern und Beobachtern kann es hilfreich sein, dass die Bewegungsgleichungen in symbolischer Form vorliegen. Hierzu wird in dieser Abhandlung ein Vorgehen beschrieben, mit dem detaillierte Mehrkörpermodelle (MKS-Modelle) mit diskretisierten flexiblen Flügeln, einfachere MKS-Modelle mit massebehafteten Scheiben als Schwenkrotoren (BILD 2), und Einmassen-Fluggerätmodelle erzeugt werden können.

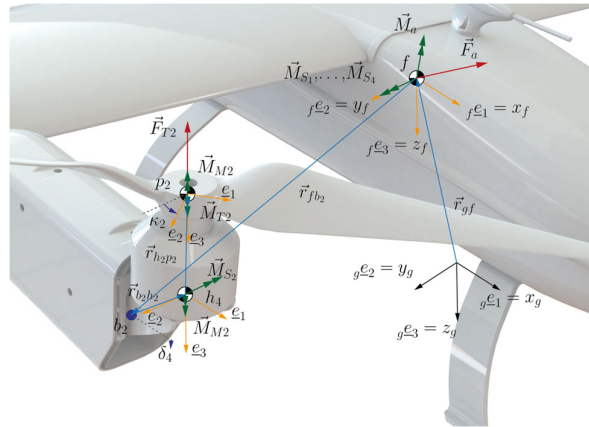


BILD 2. Vektorketten und Körper des MKS-Modells

## 2. STAND DER TECHNIK

Der Stand der Technik weist bereits einige Veröffentlichungen auf, in denen das Prinzip nach Jourdain in der Luftfahrt Anwendung findet. Ohne Anspruch auf Vollständigkeit werden nachfolgend einige genannt und eingeordnet. In [3] wird ein Vorgehen zur Modellbildung mittels Jourdain für ein Flugzeug mit zwei Antrieben beschrieben. In [4] wird ein Vorgehen zur Modellbildung mittels Jourdain für das Fahrwerk eines normalen Flugzeugs aufgezeigt. Diese Veröffentlichungen betrachten das Verhalten der Steuerbarkeit des Systems auf dem Boden. [5] und [6] beschreiben die

Modellbildung für einen kleinen Helikopter mittels des Prinzips von Jourdain. Hier fehlt der Aspekt des Tragflächenflugs in der Betrachtung. In [7] wird die Modellbildung nach Kane für ein Tiltrotor VTOL beschrieben. Mathematisch führt das Verfahren nach Kane auf dieselben Bewegungsgleichungen wie das Prinzip von Jourdain [8]. Wie in [8] beschrieben, kann das Prinzip von Jourdain einfacher auf Systeme angewendet werden.

Diese Abhandlung hebt sich gegenüber dem Stand der Technik in der Form ab, dass ein Verfahren zur automatisierten Generierung der Bewegungsgleichungen für unterschiedliche Modellierungstiefen eines Tiltrotor VTOL beschrieben wird. Das Besondere an den generierten Modellen liegt darin, dass diese auch in analytischen Gleichungen ausgeleitet werden. Darüber hinaus unterscheiden sich die hier erzeugten Modelle dadurch, dass auch Modellierungstiefen betrachtet werden, die ebenso Struktursteifigkeiten des VTOL berücksichtigen.

### 3. WERKZEUGEBENE

In diesem Kapitel werden das Vorgehen und die genutzten Grundlagen zur Modellierung von Flugsystemen vorgestellt. Darüber hinaus werden die verwendeten Lösungselemente zur Systemmodellierung und die Methodik zur Generierung der symbolischen/teilsymbolischen Differentialgleichung des Systems erläutert. Ein weiterer betrachteter Aspekt ist die automatische Generierung eines Simscape-Modells zur Validierung und Visualisierung. Es wird darüber hinaus auf die Generierung von Artefakten zur Erklärbarkeit der Systemdifferentialgleichungen eingegangen.

Um komplexe Mehrkörpersysteme modellieren zu können, ist eine durchgängige Notation zwischen mathematischer Notation und der Computernotation nötig. Hier findet die Notation aus BILD 3 Anwendung. In dem abgedruckten Beispiel wird die zeitliche Ableitung des Ortsvektors von  $f$  nach  $p2$ , der im Koordinatensystem  $b$  dargestellt ist, mittels der Rotationsmatrix  ${}^f_b A$  in das Koordinatensystem  $f$  transformiert. In BILD 2 ist beispielhaft eine Vektorkette zu den einzelnen Körpern des Mehrkörpermodells gezeigt.

Die betrachteten technischen Systeme haben die topologische Eigenschaft, dass es keine geschlossenen Wirkketten gibt. Hieraus resultiert, dass die Gelenkconfiguration einfach mittels der Minimalkoordinate  $q$  abbildbar ist. Die ersten sechs Einträge sind bei den Flugsystemen hier die sechs Freiheitsgrade im Raum.

$$(1) q = [x \ y \ z \ \varphi \ \theta \ \psi \ \dots]'$$

Zum Aufstellen der Bewegungsdifferentialgleichungen kann nun das Prinzip nach Jourdain in der Form von Gleichung (2) Anwendung finden.

$$(2) \sum_{i=1}^n g J_{Ti}^T \cdot (g \dot{p}_i - g F_i^e) + k_i J_{Ri}^T \cdot (k_i \dot{L}_i^{(C_i)} - k_i M_i^{e(C_i)}) = 0$$

Hierbei wird für  $n$  Körper des Systems je ein Summand gebildet. Die translatorische virtuelle Leistung des Körpers  $i$  wird dabei mittels der Jacobimatrix der Translation  $g J_{Ti}$  in dem Inertialkoordinatensystem  $g$  und den reaktiven Kräften  $g F_i^e$ , die auf den Körper wirken, berechnet. Die reaktiven

Kräfte lassen sich wiederum durch den Impuls  $g \dot{p}_i$  und die eingprägten Kräfte  $g F_i^e$  berechnen.

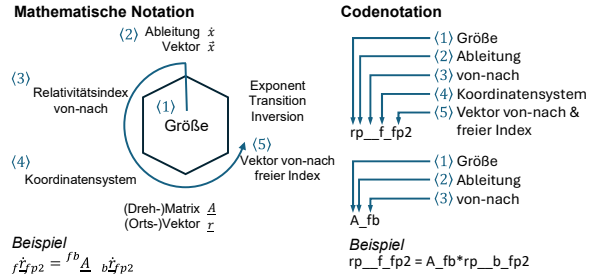


BILD 3. Notation

Für die Rotation wird die virtuelle Leistung analog in den körperfesten Koordinaten  $k_i$  berechnet. Hierbei werden der Drehimpuls  $k_i \dot{L}_i^{(C_i)}$  und die eingprägten Momente  $k_i M_i^{e(C_i)}$  bezüglich des Körperschwerpunkts ( $C_i$ ) genutzt.

Die Jacobimatrizen sind durch (3) und (4) definiert.

$$(3) g J_{Ti} = \left( \frac{\partial g^r C_i}{\partial q} \right)$$

$$(4) k_i J_{Ri} = \left( \frac{\partial g^{k_i} \omega_i}{\partial q} \right)$$

Der Impuls ist durch (5) definiert.

$$(5) g \dot{p}_i = m_i g \ddot{r}_{C_i}$$

Der Drehimpuls ist durch (6) bestimmt.

$$(6) k_i \dot{L}_i^{(C_i)} = k_i \dot{L}_i^{(C_i)} g^{k_i} \omega_i + g^{k_i} \tilde{\omega}_i k_i \dot{L}_i^{(C_i)} g^{k_i} \omega_i$$

### 3.1. Lösungselemente

Aus der topologischen Eigenschaft der offenen Gelenkketten (BILD 2) kann das System durch einen Baum beschrieben werden. Zur Beschreibung des VTOL durch ein Baumdiagramm werden generische Lösungselemente verwendet. Diese sollen die entsprechenden Größen und Teilgleichungen aus (2), (5) und (6) ermitteln. Das angewendete Prinzip liegt darin, dass aus einer Basisklasse die einzelnen Lösungselemente erben und mittels Funktionsüberladung spezifische Funktionalitäten erfüllen (BILD 4).

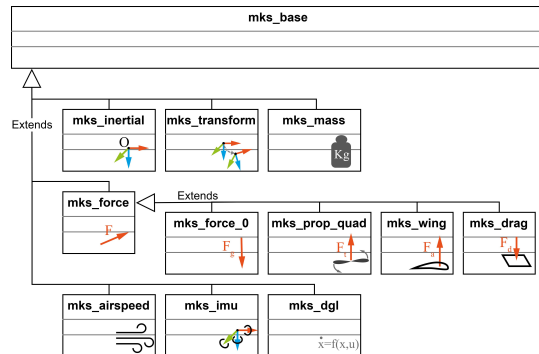


BILD 4. Klassendiagramm

Im Folgenden werden auf die implementierten Lösungselemente erläutert.

Die **Koordinatenursprungsklasse** (`mks_inertial`) stellt das Inertialkoordinatensystem  $g$  dar. In der Baumtopologie bildet ein Objekt dieser Klasse die Wurzel des

Baums. Weiter wird es im Simscape-Modell für die Parametrierung des Solvers genutzt. Es wird hier die Annahme einer flachen Erde für die Modellierung getroffen.

#### Die **Koordinatentransformationsklasse**

(mks\_transform) stellt einen Vektor und eine Koordinatenrotation zwischen den baumseitigen Vaterobjekten und den Töchterelementen dar.

Diese Koordinatentransformation kann hierbei konstant sein, durch Minimalkoordinaten beschrieben sein oder durch eine Mischung beider Varianten zustande kommen. Im Sonderfall, dass einer Koordinatentransformation kein Körper als Tochter im Baum folgt, kann auch ein Systemeingang die Transformation beschreiben. Die Klasse identifiziert die entsprechenden Freiheitsgrade und Systemeingänge und stellt diese für die Bildung des Vektors der Minimalkoordinaten bereit. Die Klasse wird dazu genutzt, die entsprechende Vektorkette von den Inertialkoordinaten zu den Töchterelementen, sowie die zugehörigen Drehmatrizen zu berechnen. Dabei wird das väterseitige Eingangskoordinatensystem als  $b_i$  und das töchterseitige Ausgangskoordinatensystem als  $f_i$  bezeichnet.

Es werden hier ebenso die Ableitungen, Winkelgeschwindigkeiten und Jacobimatrizen berechnet.

Die Klasse wird zudem genutzt, um Kräfte und Momente vom Töchterkoordinatensystem in das Vaterkoordinatensystem zu transformieren.

#### Die **Körperklasse** (mks\_mass)

dient dazu, den einzelnen Summanden aus Gleichung (2) für den jeweiligen Körper zu erstellen und schlussendlich die Summation der Teilgleichungen aus allen Töchterkörpern im Baum vorzunehmen. Die wichtigsten Eigenschaften eines Körpers sind dabei seine Masse und der Trägheitstensor. Das Eingangskoordinatensystem zeigt dabei stets auf den Masseschwerpunkt.

#### Die **Kraftklasse** (mks\_force)

wird zur Darstellung von eingepprägten Kräften und Momenten im System genutzt. Die Kräfte und Momente können dabei sowohl Funktionen aus den Minimalkoordinaten und deren Ableitungen als auch Funktionen der Systemeingänge sein. So können zum Beispiel Feder-Dämpfer-Kräfte abgebildet werden. Die Kräfte und Momente werden in den jeweiligen Väterkoordinatensystemen dargestellt. Sie wirken als eingepprägte generalisierte Kraft auf die nächsten Väterkörper im Baum. Um spezifische Kräfte am Tiltrotor VTOL aufwandsarm implementieren zu können, sind durch Vererbung noch detailliertere Kraftklassen implementiert.

#### Die **Kraftklasse in Inertialkoordinaten**

(mks\_force\_0) hat den Nutzen, die auf einen Körper wirkende Gewichtskraft in den Inertialkoordinaten zu formulieren.

#### Die **Propellerklasse** (mks\_prop\_quad)

wird genutzt, um die Schubkraft und das Widerstandsmoment eines Propellers  $i$  zu ermitteln. Die Schubkraft wird mittels des Zusammenhangs (7) ermittelt.

$$(7) F_i = K_{Ti} \omega_i^2 - |K_{Vi} [0 \ 0 \ 1] p_i v_{p_i a} \omega_i|$$

Parametriert wird die Kennlinie über den Parameter  $K_{Ti}$  für die quadratische Schubkennlinie und  $K_{Vi}$  als Abschwächungsfaktor für den einlaufenden Luftstrom in die Propellerebene. Hierbei ist  $p_i v_{p_i a}$  die Relativgeschwindigkeit zwischen dem aerodynamischen Koordinatensystem  $a$  und der Propellernabe  $p_i$  im propellerfesten Koordinatensystem  $P_i$ . Die Größe  $\omega_i$  beschreibt die Rotationsgeschwindigkeit des Propellers und lässt sich durch (8) ausdrücken.

Das Widerstandsmoment des Propellers kann mittels des Momentenparameters  $K_{Mi}$  und der Gleichung (9) berechnet werden.

$$(8) \omega_i = [0 \ 0 \ 1] p_i \omega_{g p_i}$$

$$(9) M_i = K_{Mi} \omega_i^2$$

#### Die **Flügelementklasse** (mks\_wing)

dient zur Berechnung der Auftriebs- und Widerstandskräfte eines Tragflächenelements, sowie des an der Tragfläche wirkenden Moments. Hierbei wird im Gegensatz zu [9] nicht das vollständige System betrachtet, es findet vielmehr eine Zerteilung in einzelne Flügelementstücke zur Modellbildung statt BILD 10. Die Berechnung der Kräfte und Momente wird wie in [10, p. 442] beschrieben durchgeführt.

Es ergibt sich für die Kraft am  $i$ -ten Flügelement der linken Haupttragfläche  $wLi$  der funktionale Zusammenhang (10). Dabei stellt  $\eta_{wLi}$  den Klappenwinkel dar,  $c_{L,wLi}$  die Auftriebskennlinie, und  $c_{D,wLi}$  die Widerstandskennlinie. Die Parameter  $A_{wLi}$ ,  $\lambda_{k,wLi}$ ,  $l_{\mu,wLi}$ ,  $l_{Flap,wLi}$ ,  $\kappa_{wLi}$  sind geometrische Parameter und Korrekturfaktoren zur Berechnung der Kräfte.

$$(10) {}_{wLi} F_{wLi}^e = f \left( {}_{wLi} v_{wLi a}, \eta_{wLi}, \rho, c_{L,wLi}, c_{D,wLi}, \dots \right)$$

Die Momente können analog mit der Funktion (11) berechnet werden.

$$(11) {}_{wLi} M_{wLi}^e = f \left( {}_{wLi} v_{wLi a}, \eta_{wLi}, \rho, c_{M,wLi}, \dots \right)$$

Die Kräfte und Momente werden hierzu in den tragflächenelementspezifischen Koordinaten dargestellt. Zur Berechnung der Anströmgeschwindigkeit und des Anstellwinkels werden dabei nur die Vektorkomponenten in der xz-Ebene des Tragflächenelements berücksichtigt. Durch die Unterteilung der Tragfläche in kleine Teilstücke wird beispielsweise die aerodynamische Dämpfung durch die unterschiedlichen Geschwindigkeiten der Tragflächenelemente beim Rollen des Systems berücksichtigt. In [11] wurde bereits ein ähnlicher Ansatz zur elementbasierten Berechnung der Aerodynamik beschrieben. In diesem Fall werden jedoch die Koordinatentransformationen durch die Baumstruktur aus dem Element in die Gesamttopologie verlagert. Es findet eine stärkere Funktionstrennung statt. Der aerodynamische Einfluss zwischen den Elementen wird in der Flügelementklasse vernachlässigt.

#### Die **Widerstandselementklasse** (mks\_drag)

berechnet einen aerodynamischen Ersatzwiderstand für beispielsweise den Rumpf des VTOL. Hierzu wird die Theorie der schräg angeströmten Platte verwendet. Wobei auch hier, wie beim Flügelement nur die xz-Ebene des Elements berücksichtigt wird. Die Widerstandskennlinie des Rumpfelements  $Di$  wird durch (12) abgebildet. Der Auftrieb der Platte wird nicht berücksichtigt.

$$(12) c_{D,Di} = 0.05 + 1.18 \sin \alpha_{Di}^2$$

Die **Sensorelemente** dienen zur Implementierung von zusätzlichen Systemausgängen, welche nicht direkt die Zustandsgrößen des Systems sind, sondern durch Ausgangsgleichungen aus den Zustandsgrößen indirekt berechnet werden.

#### Die **Inertiale Messeinheitsklasse** (mks\_imu)

dient zur Ausleitung und Berechnung der typischen Größen einer IMU. Die Lage und Orientierung der IMU im System wird durch das Väterkoordinatensystem definiert. Darüber hinaus wird auch die entsprechende Position des Elements in globalen Koordinaten dem Systemausgang hinzugefügt. Es können durch die eindeutige Benennung des Elements auch mehrere IMU-Elemente im System verbaut werden.

Die **Anströmsensorklasse** (`mks_airspeed`) berechnet die Relativgeschwindigkeit der Luft gegenüber dem Sensor in den elementfesten Koordinaten. Hierbei wird die Relativgeschwindigkeit in den Komponenten des Koordinatensystems zu den Systemausgängen hinzugefügt.

Die **Zusatzdynamikkasse** (`mks_dgl`) wird zur Abbildung von weiteren Teilsystemdynamiken des VTOLs verwendet. Dies können beispielsweise elektrische Teilsysteme mit eigenen Dynamiken, wie Batterien, Antriebsmotoren oder Servoantriebe sein. Hierrunter fallen auch die I-Anteile der Regler. Die Klasse wird mit einem Vektor an Zustandsgrößen und der Differentialgleichung instanziiert. Die topologische Einordnung im Baum ist für die Funktion nicht von Relevanz, für die Steigerung der Erklärbarkeit bietet sich eine Einordnung in der Nähe der zugehörigen Lösungselemente des Mehrkörpersystems an.

### 3.2. Vorgehen zur Generierung der Differentialgleichungen des Mehrkörpersystems

Die spezifische Generierung der Modellierungstiefe des VTOL-Modells wird in Kapitel 4 beschrieben. Liegt die spezifische Topologie aus parametrisierten Objekten in Form des Baums vor, so können durch eine Abfolge an rekursiven Funktionsaufrufen die Bewegungsgleichungen erzeugt werden. Die benötigten Funktionen sind in der Basisklasse (`mks_base`) bereits angelegt und werden bei Bedarf durch Überschreiben um spezifische Funktionalität ergänzt.

Es wird hier zwischen zwei Arten der Rekursion unterschieden. In der *Rekursion-A* wird zunächst die entsprechende Funktion des jeweiligen Objektes aufgerufen und erst dann die Funktionen der Tochterobjekte aufgerufen. Es werden die Funktionen aus der Richtung der Baumwurzel kommend, in Richtung dessen Blättern fortschreitend ausgeführt. Dieser rekursive Aufruf ist so gestaltet, dass der aufgerufenen Funktion Zeiger auf die Väterobjekte übergeben werden. Über diesen Zeiger ist der Zugriff auf Größen des Väterobjekts möglich. Die *Rekursion-B* ruft erst die Funktionen der Tochterobjekte auf und erst nach dessen Ausführung wird die Funktion des eigenen Objektes ausgeführt. Es werden die Funktionen aus Richtung der Blätter des Baums hin zur Wurzel ausgeführt. Dieser rekursive Aufruf ist so gestaltet, dass der aufgerufenen Funktion Zeiger auf die Tochterobjekte übergeben werden. So kann durch die aufgerufene Funktion auf Größen der Tochterobjekte zugegriffen werden.

#### 0a) Analyse massebehafteter Körper

Durch die *Rekursion-B* wird analysiert, welchen Objekten ein massebehafteter Körper als Tochterelement im Baum folgt. Dieses Merkmal wird in den Objekten abgelegt.

#### 1a) Ermittlung der Minimalkoordinaten

Der Vektor der Minimalkoordinaten  $\mathbf{q}$  des Mehrkörpersystems wird durch *Rekursion-B* erzeugt. Hierzu findet in den Objekten der Koordinatentransformationsklasse eine Analyse der Parametrierung statt. Es wird geprüft, ob die jeweilige Richtung oder Rotation der Transformation von zeitlich variablen Größen abhängt und der Transformation ein Körper als Tochterelement folgt. Ist dies der Fall, so ist die zeitlich variable Größe Teil der Minimalkoordinaten.

#### 2a) Ermittlung der Systemeingänge

Durch die *Rekursion-B* kann nun der Vektor der Systemeingänge  $\mathbf{u}$  ermittelt werden. Systemeingänge stellen hierbei alle bisher nicht definierten symbolischen Variablen der Parametrierung der Elemente dar. Je nach zu untersuchender

Fragestellung können dies neben klassischen Systemeingängen auch Designparameter wie Federsteifigkeiten oder Dämpfungen des Systems sein, die in verschiedenen Simulationen variiert werden.

#### 3a) Berechnung der Vektorketten

Mittels der *Rekursion-A* werden die Vektorketten zu den einzelnen Objekten im Baum aufgestellt (BILD 2). Dazu wird der Zusammenhang (13) genutzt.

$$(13) {}^g\mathbf{r}_{gf_i} = {}^g\mathbf{r}_{gb_i} + {}^{gb_i}A_{b_i} \mathbf{r}_{b_if_i}$$

Aus den Ortsvektoren, dargestellt im Inertialkoordinatensystem, kann durch zeitliche Differentiation der symbolischen Gleichungen die Geschwindigkeit (14) und Beschleunigung (15) des entsprechenden Punktes ermittelt werden.

$$(14) {}^g\mathbf{v}_{gf_i} = {}^g\dot{\mathbf{r}}_{gf_i}$$

$$(15) {}^g\mathbf{a}_{gf_i} = {}^g\ddot{\mathbf{r}}_{gf_i}$$

Die Berechnung der Rotationsmatrix des Ausgangskoordinatensystems  $f_i$  kann mittels (16) erfolgen.

$$(16) {}^{gf_i}A = {}^{gb_i}A_{b_i} {}^{b_i}A_{f_i}$$

Die Berechnung der Winkelgeschwindigkeit erfolgt anschließend mittels (17).

$$(17) {}^{gf_i}\tilde{\boldsymbol{\omega}}_{f_i} = {}^{gf_i}A' {}^{gf_i}\dot{A}$$

Abschließend wird die zeitliche Differentiation der Winkelgeschwindigkeit  ${}^{gf_i}\tilde{\boldsymbol{\omega}}_{f_i}$  durch symbolische Differentiation ermittelt. Somit liegen nach der Rekursion für jeden Knotenpunkt im System der Ortsvektor, die Geschwindigkeit und Beschleunigung als Funktion der Minimalkoordinaten vor. Für die rotatorischen Größen ist dies mit  ${}^{gf_i}A$ ,  ${}^{gf_i}\tilde{\boldsymbol{\omega}}_{f_i}$  und  ${}^{gf_i}\dot{\tilde{\boldsymbol{\omega}}}_{f_i}$  ebenso gegeben.

#### 4a) Berechnen der eingepprägten Kräfte

Mittels der *Rekursion-B* werden nun im System die auf die jeweiligen Körper wirkenden eingepprägten Kräfte und Momente berechnet. Zum Beispiel werden zur Berechnung der Auftriebs- und Widerstandskraft eines Flügelements die nun im Element vorliegenden Geschwindigkeitszusammenhänge aus (14) verwendet, um die Anströmgeschwindigkeit und den Anstellwinkel des Elements zu berechnen.

Zur weiteren Verwendung der eingepprägten Kräfte und Momente in der Systemtopologie werden die Kräfte und Momente durch die Rekursion aufsummiert und im Fall einer Koordinatentransformation aus dem Ausgangskoordinatensystem  $f_i$  in das Eingangskoordinatensystem  $b_i$  transformiert (siehe Gleichung (18) und (19)).

$$(18) {}_{b_i}F_i^e = {}^{b_if_i}A_{f_i} F_i^e$$

$$(19) {}_{b_i}M_i^{e(b_i)} = {}^{b_if_i}A_{f_i} M_i^{e(f_i)} + {}_{b_i}r_{b_if_i} \times {}_{f_i}F_i^e$$

In einem Körperelement liegen somit nun die Summanden der eingepprägten Kräfte  ${}_{k_i}F_i^e$  und Momente  ${}_{k_i}M_i^{e(c_i)}$ , die auf eben jenen wirken, vor (siehe (2)). Die eingepprägten Kräfte wirken per Definition nur den Körper, an dem sie angreifen. Der Körper, an dem die summierten, eingepprägten Kräfte wirken, ist immer der nächste Körper in Richtung der Wurzel des Baums. Daher werden die eingepprägten Kräfte für die weitere Rekursion durch die Körperklasse als Nullvektor an das jeweilige Väterelement weitergegeben.

#### 4b) Aufstellen und Summieren der Teilsommanden

In den Körperelementen liegen nun alle mathematischen Größen vor, um den jeweiligen Summanden aus Gleichung (2) zu berechnen. Hierrunter fallen die auf das jeweilige Element wirkenden eingepprägten Kräfte und Momente und die Lage und Orientierungsgrößen sowie deren Ableitungen. Zur Berechnung der Teilsommanden, werden zunächst der Impuls- und der Drehimpuls durch (5) und (6)



bestimmt und anschließend die Jacobimatrizen mittels (3) und (4) ermittelt. Mittels der *Rekursion-B* werden diese Summanden über den Baum hinweg summiert und liegen anschließend in der Wurzel als Gesamtsumme vor.

#### 5a) **Aufstellen der Differentialgleichung des MKS**

Das zu Null setzen der in der Wurzel des Baums vorliegenden Summe aus 4b) erzeugt nun die Bewegungsdifferentialgleichung nach Jourdain aus Gleichung (2).

#### 6a) **Ausleiten von zusätzlichen Systemausgängen**

Um neben den Zuständen des Systems noch weitere Ausgangsgleichungen definieren zu können, wird mittels der *Rekursion-B* ein weiterer Vektor mit Systemausgangsgleichungen gebildet. Hier können Größen der IMU oder der Pitotsonde enthalten sein.

#### 7a) **Reduktion der Differentialgleichungsordnung**

Um ein einheitliches Simulinkmodell zur Simulation der Differentialgleichung nutzen zu können, wird diese auf eine ODE erster Ordnung reduziert. Darüber hinaus werden die neuen Zustandsgrößen  $x$  nach dem Benennungsschema auf BILD 3 benannt.

#### 7b) **Massen-Matrix-Form**

Zur Verwendung wird die ODE erster Ordnung nun in die Massen-Matrix-Form überführt. Anschließend werden die Matrizen als Matlabfunktionen ausgeleitet. Es entstehen die Funktionen  $\text{ODE\_M}(t,x)$ ,  $\text{ODE\_F}(t,x,u)$  und  $\text{ODE\_G}(t,dx,x,u)$ . Mit diesen kann mittels der Gleichung (20) das neue  $dx$  berechnet werden.

$$(20) \, dx = \text{ODE\_M}(t,x) \cdot \text{ODE\_F}(t,x,u)^{-1}$$

Der Ausgang des Systems ist durch (21) definiert.

$$(21) \, y = \text{ODE\_G}(t, dx, x, u)$$

Nun kann das System mittels des Simulationsmodells (BILD 5) simuliert werden.

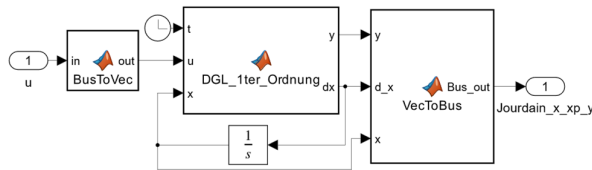


BILD 5. Simulationsmodell

### 3.3. Vorgehen zur Ergänzung um die Differentialgleichungen der Zusatzdynamiken

In diesem Kapitel wird die Erweiterung des Vorgehens aus dem vorangegangenen Kapitel um die zusätzlichen Systemdynamiken beliebiger Ordnung erläutert. Dazu werden die folgenden Prozessschritte im vorangegangenen Vorgehen ergänzt.

#### 1b) **Zusätzliche Zustandsgrößen ermitteln**

Der Gesamtvektor der Zustandsgrößen  $x_{rest}$ , der die Dynamiken aus den Objekten der Zusatzdynamikkategorie enthält, wird durch *Rekursion-B* erzeugt.

#### 4c) **Vektor mit Zusatzdifferentialgleichungen**

Mittels der *Rekursion-B* wird der Satz der Zusatzdifferentialgleichungen erzeugt. Alle Zusatzdifferentialgleichungen liegen schlussendlich in der Wurzel des Baums vor.

#### 5b) **Zusammenführen der Zustandsvariablen**

Die Minimalkoordinaten  $q$  und  $x_{rest}$  bilden zusammen einen Gesamtvektor.

#### 5c) **Zusammenführen der Differentialgleichungen**

Die Differentialgleichungen aus dem MKS-System werden mit den Zusatzdifferentialgleichungen in einem Vektor zusammengeführt.

Durch diese Erweiterung können nun auch nicht aus der Mehrkörperdynamik stammende Systemdynamiken simuliert werden.

### 3.4. Generierung eines topologischen Modells in Simscape

Zur Modellvalidierung mittels Simulation wird zusätzlich zu dem analytisch aufgestellten Modell ein Zwillingsmodell in Simscape automatisiert erstellt. Dieses Simscape Modell dient auch der anschaulichen dreidimensionalen Visualisierung des Systemverhaltens mittels des Simscape Multi-body Explorers. So können Parametrierungsfehler der Koordinatentransformationen einfacher identifiziert werden.

Die Generierung des Simscape Modells wird ähnlich zur Generierung des analytischen Modells durch rekursive Funktionsaufrufe über die Elemente im Baum durchgeführt. Es wird zunächst die Größe der einzelnen Systemblöcke ermittelt. Hierbei sind die Anzahl der Eingangs- und Ausgangsgrößen der einzelnen Lösungselemente maßgebend für die Blockhöhe. Anschließend werden die Lösungselemente aus einer Bibliothek BILD 6 entnommen, im Baum platziert, parametrisiert und entsprechend der Parametrierung angepasst.

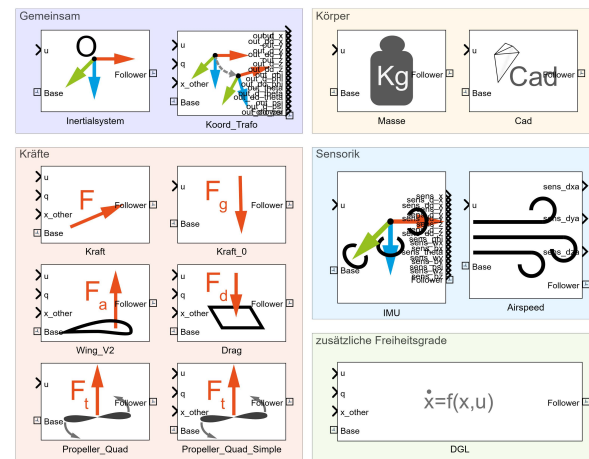


BILD 6. Simscape Bibliothek

Hierzu wird die *Rekursion-A* verwendet. Der Base-Konnektor des jeweiligen Blocks wird dabei über eine ungerichtete Verbindung mit dem Follower-Konnektor des Väterelements verknüpft. Neben dieser Verbindung kann ein Block weitere Eingänge aufweisen. Dies können der Systemeingang  $u$ , Minimalkoordinaten  $q$  und deren Ableitungen  $\dot{q}$ ,  $\ddot{q}$  sowie zusätzlichen Systemzustände  $x_{rest}$  und deren Ableitung  $\dot{x}_{rest}$  sein. Die Elementeingänge werden bis auf den Systemeingang  $u$  durch From-Blöcke realisiert (BILD 7).

Die Blockausgänge variieren je nach Elementklasse. Allgemein wird ein Ausgang mit einem spezifischen Datum des Ausgangsbusses verbunden und mit einem GoTo-Block mit lokaler Sichtbarkeit zur weiteren Verwendung im Simscape Modell verknüpft. Bei Koordinatentransformationen mit Gelenken stellen die Minimalkoordinaten und deren

Ableitungen die Blockausgänge dar. Im Fall der Zusatzdynamikkasse sind die Ausgänge durch die in den Differentialgleichungen verwendeten Systemzustände und deren Ableitungen definiert. Die Sensorelemente haben ihre anwendungsspezifischen Ausgänge.

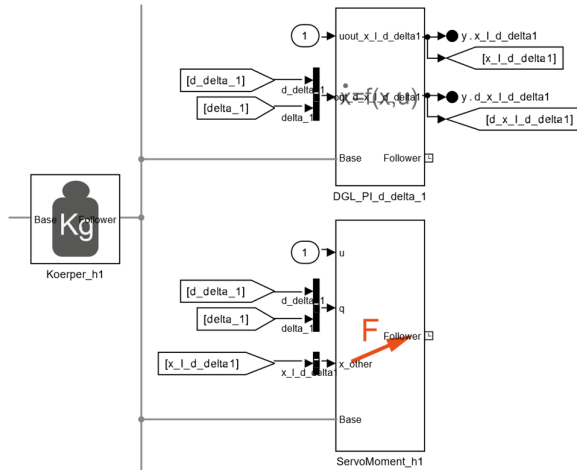
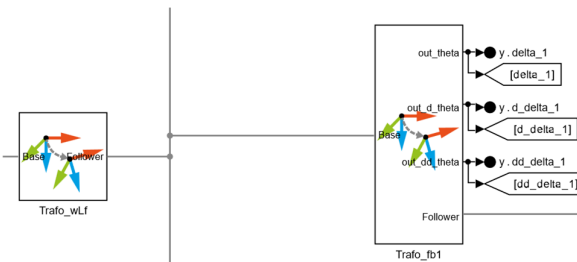


BILD 7. Ein- und Ausgänge

Um die Zusatzdifferentialgleichungen, die Krafteingänge und die Koordinatentransformationen mit den jeweiligen spezifischen symbolischen Ausdrücken entsprechend der Parametrierung abbilden zu können, werden für diese objektspezifische Matlabfunktionen ausgeleitet. Dies können beispielsweise die Kraftgleichung eines Feder-Dämpfer-Elements sein, aber auch Matlabfunktionen für die Umsetzung einer Zusatzdifferentialgleichung.

Die Koordinatentransformationen werden entsprechend der Parametrierung angepasst. In BILD 8 sind eine statische Koordinatentransformation (Trafo\_wLf) und eine Transformation mit Gelenk um die y-Achse mit der zugehörigen Minimalkoordinate  $\delta_1$  zu sehen (Trafo\_fb1). Hierzu wird das Innere des Blockes entsprechend der Parametrierung modifiziert. Es werden zum Beispiel nicht benötigte Ausgänge entfernt.



## BILD 8. Koordinatentransformationen

Das generierte Simscape Modell unterscheidet sich in Bezug auf das analytisch generierte Modell nur in der Art des verwendeten Lösungsalgorithmus der Gleichungen. Bei dem analytischen Modell wird der Euleralgorithmus zur Simulation verwendet. Da das Simscape Modell potenziell ein differential-algebraisches Gleichungssystem (DAE-System) erzeugt verwendet es zunächst das Verfahren aus [12] um ein ODE-System zu generieren. Anschließend wird das Modell simuliert.

### 3.5. Erklärbarkeit der Systemtopologie

Um die Erklärbarkeit des Generats auch für das aus symbolischen Gleichungen bestehende Modell für den Nutzer zu gewährleisten, wird durch die Werkzeugebene noch eine ASCII-Grafik der Baumtopologie des Systems erzeugt (BILD 9). Auch dieses Generat wird mittels rekursiver Funktionsaufrufe im Baum und Funktionsüberladung der einzelnen Lösungselemente erzeugt. Die Komponenten werden dabei unter Berücksichtigung des Namens und des Typs, sowie der Eingänge und der Freiheitsgrade des Lösungselements abgedruckt. So entsteht eine menschenslesbare Repräsentation der Topologie.

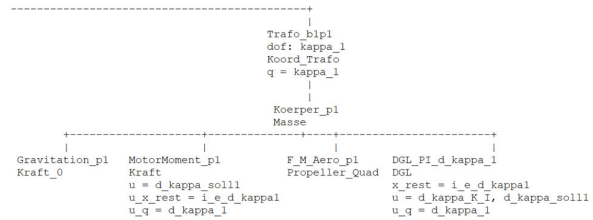


BILD 9. Ausschnitt des ersten Propellers aus der Systemtopologie in der ASCII-Repräsentation

#### 4. ANWENDUNGSEBENE

Um eine durchgängige Modellierung der Tiltrotor VTOL in unterschiedlichen Modellierungstiefen zu erreichen, stellt die Anwendungsebene ein Vorgehen zur Generierung der Systemtopologien zur Verfügung. Darüber hinaus werden weitere Funktionen zur Abstraktion von wiederkehrenden Parametrisierungen der Systemelemente eingeführt. In der Anwendungsebene wird hierzu eine fluggerätespezifische Beschreibungsdatei genutzt, um für die verschiedenen Detaillierungsebenen aus den Klassen der Werkzeugebene spezifische parametrisierte Objekte zu instanzizieren. Durch die Beschreibungsdatei können verschiedene Tiltrotor VTOL Konfigurationen mit unterschiedlicher Größe, Anzahlen an Propellern oder verschiedenen Tragflächenkonfigurationen beschrieben werden. Die parametrisierten Objekte werden in einem Baumgrafen abgelegt. Der Graf entspricht in seiner Topologie der jeweiligen Modellierungstiefe des zu generierenden Tiltrotor VTOL Modells.

#### 4.1. Modellierungstiefen

Durch die unterschiedlichen Anforderungen an Simulationsmodelle, durch verschiedenste Synthesaufgaben im mechatronischen Systementwurf, werden Modelle mit veränderlicher Modellierungstiefe benötigt. In diesem Kapitel wird ein Überblick über die implementierten Modellierungstiefen vermittelt.

### 1) **Einmassenmodell ohne Aktordynamik**

Hierbei wird der Rumpf zusammen mit den Tragflächen und den Antriebssystemen als ein Körper modelliert. Die durch den Regler geforderten Motordrehzahlen, Schwenkwinkel und Kappenstellungen liegen direkt an. Die Auftriebskräfte der Tragflächen werden durch segmentierte Tragflächenteilstücke berechnet. Hierdurch wird die aerodynamische Dämpfung des Systems bereits mitberücksichtigt. Dies ist die Modellierungstiefe mit dem größten Grad der Abstraktion, die hier umgesetzt wurde.

2) **Einmassenmodell mit PT1-Dynamik für Propellerdrehzahlen und Schwenkwinkel**

Dieses Modell erweitert die vorangegangene Modellierungstiefe um PT1-Dynamiken für die Propellerdrehzahlen und die Schwenkwinkel der Antriebssysteme.

3) **Einmassenmodell mit PT1-Dynamik für Propellerdrehzahlen und PT2-Dynamik für Schwenkwinkel und Klappenstellungen**

Dieser Abstraktionsgrad erweitert das Einmassenmodell mit PT2-Dynamiken für alle unterlagerten lagegeordneten Größen. Dies ist physikalisch durch die zwei Integratoren in der Modellierung der Position einer Masse begründet.

4) **Mehrkörpermodell mit massebehafteten Propellern (Annahme: Trägheit der Propeller ist die einer ausgedehnten Scheibe)**

In dieser Modellierungstiefe, wird bei einem System mit vier Propellen, ein Mehrkörpermodell mit fünf Körpern erzeugt. Die Propeller werden mit der Trägheit einer ausgedehnten Scheibe versehen. Diese Vereinfachung führt dazu, dass größere Schrittweiten zur Simulation verwendet werden können. Der Grund hierfür liegt in den schwenkbaren Propellersystemen. Bezogen auf das Koordinatensystem in der Schwenkachse des Systems gibt es keine Variation der Trägheit um die Schwenkachse bei Rotation der Propeller. Die Dynamiken der Steuerklappen werden weiter durch Verzögerungsglieder zweiter Ordnung modelliert.

Um in der Detaillierungsebene weiterhin die Sollwerte für die Propellerdrehzahl und den Schwenkwinkel als Systemeingang nutzen zu können, ist eine unterlagerte Regelung des Antriebsmomentes für das Servomotor der Schwenkmechanik und den Antriebsmotor nötig.

Für die Schwenkmechanik wird hierzu ein kaskadierter P-PI-Regler eingesetzt. Dieser Regler nutzt wiederum Saturierungen für das maximale Moment und die maximale Schwenkrate. Somit sind die Schwenkaktoren im Moment und der Geschwindigkeit begrenzt. Für die Propellerdrehzahl wird ein PI-Regler genutzt.

Somit kann die gleiche Schnittstelle zum Fluglageregler in allen Modellierungsgraden eingesetzt werden.

5) **Mehrkörpermodell mit massebehafteten Schwenkmechanismen und Propellern (Annahme: Trägheit der Propeller ist die einer ausgedehnten Scheibe)**

Die vorangegangene Modellierungstiefe wird hier weiter um massen- und trägheitsbehaftete Körper der Schwenkmechaniken ergänzt.

6) **Mehrkörpermodell mit massebehafteten Schwenkmechanismen und Propellern**

In dieser Modellierungstiefe wird auf die Vereinfachung des Trägheitstensors der Propeller verzichtet. Zur Simulation sind hier kleinere Schrittweiten nötig. Insbesondere zweiblättrige Propeller stellen hier wegen der

großen Varianz zwischen  $\mathbf{I}_{P_i,xx}^{(C_{P_i})}$  und  $\mathbf{I}_{P_i,yy}^{(C_{P_i})}$  eine besondere Herausforderung an die Bandbreite des Schwenkwinkelreglers dar. Dieser Effekt kann bei der mechanischen Auslegung des Schwenksystems und der Parametrierung der Schwenkantriebe nicht vernachlässigt werden. Mögliche Fehlerbilder sind hier defekte Zahnräder in den Servomotoren, ungewollte durch den Servoantrieb induzierte Schwingmodi oder auch defekte mechanische Komponenten des Schwenksystems.

7) **Mehrkörpermodell mit Steifigkeiten in den Flügelwurzeln und massebehafteten Tragflächen, Schwenkmechanismen und Propellern (Annahme: Trägheit der Propeller ist die einer ausgedehnten Scheibe)**

Ziel dieser Modellierungstiefe ist es die Einflüsse der Steifigkeiten in den Tragflächen auf die Fluglageregelung zu untersuchen. Zur Modellierung werden hier masse- und trägheitsbehaftete Körper für die Tragflächen ergänzt. Diese sind wiederum durch rotatorische Gelenke mit dem Rumpf verbunden. Die Gelenke werden durch Feder-Dämpfer Momente in der Baumtopologie ergänzt. Dabei ist das Gelenk um die z-Achse für die Modellbildung aufgrund der großen Steifigkeit und der daraus resultierenden großen Eigenfrequenz nicht relevant für die Fluglageregelung und findet keine Berücksichtigung. Es werden die Freiheitsgrade um die x-Achse und die y-Achse berücksichtigt und durch die Minimalkoordinaten  $\sigma_{wi}$  und  $\tau_{wi}$  abgebildet.

## 4.2. Vorgehen der Modellbildung in der Anwendungsebene

Für die einzelnen Detaillierungsgrade der Modellierungstiefe wird für ein Fluggerät eine gemeinsame Parameterdatei genutzt. In dieser werden alle relevanten Parameter des Tiltrotor VTOL's vorgehalten. Zur Generierung der detaillierungsgrad spezifischen Baumtopologie werden zunächst symbolische Variablen für alle relevanten Parameter und Freiheitsgrade generiert. Alle durch die Parameterdatei bekannten Größen werden in dem Satz der symbolischen Variablen anschließend substituiert. So kann neben einem vollständig parametrisierten Modell, auch ein vollständig aus symbolischen Variablen bestehendes Modell oder beliebige Mischformen generiert werden. Durch diese Möglichkeit können mit geringem Aufwand Parameterstudien durchgeführt werden, aber auch Modelle mit vollständiger Parametrierung für die Reglersynthese erzeugt werden.

Zur Generierung sich wiederholender Teilsysteme stellt die Applikationsebene Funktionen bereit, die die Erstellung und Parametrierung der Basisobjekte vornimmt. Dies sind zum Beispiel die Funktionen zur Generierung der Schwenkrotoren oder der Tragflächen des VTOL's. Es wurden jedoch auch einfachere Funktionen zur Erzeugung der PD-/PID-Regler oder PTx-Dynamiken mit Saturierungen umgesetzt. Diese instanzieren die Objekte der Zusatz-DGL-Klasse entsprechend.

Nach der Instanziierung der Objekte der Lösungselementklassen werden die generierten Objekte durch Funktionen wiederum topologisch in der Gesamtsystemtopologie eingeordnet.

Im Anschluss werden einheitliche Ein- und Ausgangsbusse zum einfachen Austausch der Systemmodelle der verschiedenen Detaillierungsstufen geschaffen. Neben den Bussen wird eine Funktion zur Zuordnung der Größen in den Eingangsvektor des Systems erzeugt. Ebenso wird eine Funktion zur Zuordnung der Größen des Ausgangsvektors in den Ausgangsbus erzeugt. Diese Busse sind für ein Fluggerät über die Modellierungstiefen einheitlich, um einen einfachen Modellaustausch während der Reglersynthese und -validierung zu ermöglichen.

Die Anwendungsebene generiert für die Modelle Simulink Data Dictionaries. Diese beinhalten alle benötigten Parameter und Busdefinitionen. Es wird für die Sätze aus Parametern und Busdefinitionen unterschieden, ob ein Modell

nur aus symbolischen Gleichungen, ausschließlich ein Simscapemodell oder beides eingesetzt werden soll. Somit sind die Data Dictinaies auf die für den Modelltyp minimale Größe reduziert.

#### 4.3. Implementierung der flexiblen Flügelemente

Dieses Kapitel beschreibt die Umsetzung von flexiblen Flügelementen durch die Anwendungsebene. Um bedarfsorientiert verschiedene Steifigkeiten einbringen zu können, sind die Steifigkeitselemente in der Beschreibungsdatei so umgesetzt, dass zu jedem Steifigkeitselement ein Vektor an Systemkomponenten angegeben werden kann, der definiert welche weiteren Teilsysteme durch die Steifigkeit angebunden sind. Hierrunter können wiederum auch weitere Steifigkeitselemente fallen. So können zum Beispiel verschiedene hintereinander folgende Steifigkeiten in einem Flügel abgebildet werden. Im Fall eines VTOLs mit acht Propellern kann dies zum Beispiel eine Steifigkeit in der Flügelwurzel sein, eine weitere Steifigkeit nach dem ersten Motorträger und eine letzte Steifigkeit vor den Querrudern. Eine weitere Möglichkeit besteht darin, eine Steifigkeit für das Höhenleitwerk und Seitenleitwerk abzubilden.

Für die Reglersynthese des Albacopters 0.1 sind die Steifigkeiten in den Haupttragflächen von Relevanz. Dies ist auf die Anbindung der Antriebssysteme über die Motorträger an der Haupttragfläche zurückzuführen. Diese Anbindung resultiert in langen Hebelarmen zwischen den Antriebssystemen und den Tragflächen des VTOL's. Je nach Reglerparametrierung und struktureller Auslegung des Systems können hier flugdynamisch relevante Schwingungsmodi auftreten.

Um die Simulationsdauern gering zu halten, wird zunächst für das Modell des Albacopter 0.1 mit vier Schwenkrotoren nur je eine Steifigkeit mit zwei Freiheitsgraden in der Flügelwurzel der linken und rechten Tragfläche vorgesehen. Hierdurch können die beiden auf dem Prüfstand als relevant identifizierten Schwingungen simuliert werden. Bei dem ersten relevanten Modus handelt es sich um eine gegenphasige Nickschwingung des Rumpfes gegenüber der Motorträger. Der zweite beobachtete Modus ist eine auf und ab Bewegung der Motorträger gegenüber dem Rumpf.

#### 4.4. Nachverfolgbarkeit, Reproduzierbarkeit und Dokumentation

Bei der Reglersynthese von Flugsystemen ist es von besonderer Relevanz die Nachverfolgbarkeit und Reproduzierbarkeit der Systemmodelle sicherzustellen. Dies wird durch eine Git Integration erreicht. Hierbei wird sichergestellt, dass für jedes generierte Modell der Generator versioniert ist, sowie der entsprechende Commit-Hash des Generators im Generat abgelegt wird. Die verschiedenen Fluggeräte und Modellierungstiefen werden durch verschiedene Branchbezeichnungen in Git nachgehalten. Die Detaillierungsebenen lassen sich so durch einen Branchwechsel des Modells auswählen. Das Systemmodell kann über ein Submodul in das Gesamtmodell integriert werden.

Die Erklärbarkeit der Modelle wird durch das Ablegen der Fluggerätparameter in einem JSON-Format gewährleistet. Änderungen in den Parametern werden durch die Git Änderungsverfolgung offen dem Generatsnutzer dargelegt.

Ebenso wird dieses Vorgehen für die Einstellungen der Detaillierungsebene verwendet. Hier wird beispielsweise abgelegt, ob massenbehaftete Propeller in der Detaillierungsebene genutzt werden sollen. Die ASCII-Repräsentation des Modells wird zur Sicherung der Erklärbarkeit ebenso mit im Generat abgelegt. So kann auch dem Modellnutzer des analytisch generierten Modells ein hoher Grad an Transparenz ermöglicht werden.

### 5. ERGEBNISSE

Aus dem Vorgehen zur automatischen Modellgenerierung entsteht für den Detaillierungsgrad mit Steifigkeiten in den Flügeln das in BILD 11 abgedruckte Gesamtsystemmodell. Das Simscape-Mehrkörpermodell enthält dabei die in BILD 10 gezeigte 3D-Visualisierung der generierten Funktionselemente. So kann dem Nutzer ein visueller Eindruck des Systemverhaltens und des generierten Systems vermittelt werden.

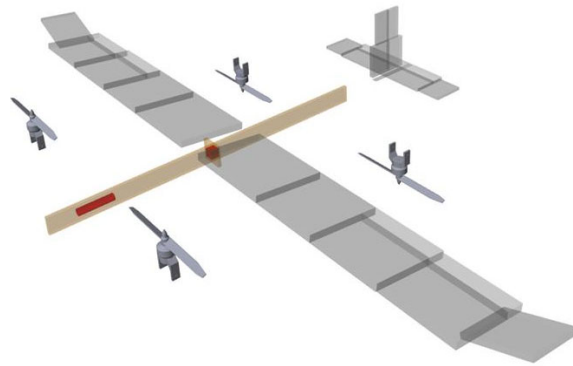


BILD 10. Visualisierung der physikalischen Repräsentation der Elemente des Albacopter® 0.1

Durch die Linearisierung der verschiedenen Modellierungstiefen des analytisch generierten Modells kann der Einfluss der Modellierungstiefen deutlich gemacht werden. In BILD 12 sind die Pole und Nullstellen einiger Modellierungsebenen für den Start- und Landevorgang des Albacopter 0.1 abgedruckt. Es wird ersichtlich, dass mit steigender Modellierungstiefe weitere Pole und Nullstellen im linearisierten System hinzukommen. Die Starkkörpermoden des Systems bleiben weitestgehend unverändert.

### 6. ZUSAMMENFASSUNG UND AUSBLICK

In der Abhandlung wurde gezeigt, wie für die Fluggerätklasse der Tiltrotor VTOL mittels eines generischen Vorgehens Modelle unterschiedlicher Modellierungstiefen generiert werden können. Hierbei sorgt eine einheitliche Beschreibungsdatei des Fluggeräts dafür, dass sowohl Mehrkörpermodelle nach dem Verfahren von Jourdain wie auch topologisch identische Simscape Modelle generiert werden können. Die analytischen Modelle bieten sich gegenüber den topologisch erzeugten Modellen für den Einsatz als Modell im Regler oder Beobachtern an. Sowohl die analytischen Modelle als auch die Simscape Modelle können bei der Optimierung der Reglerparameter genutzt werden [13].

Das Vorgehen ließe sich um weitere Lösungselemente erweitern. Hier bietet sich die Erweiterung um ein Batteriemodell durch ein einfaches Thévenin-Modell an [14]. Dieses



9

könnte durch Vererbung aus der Zusatzdynamikkategorie erzeugt werden. Ähnliches kann für Motormodelle erweitert werden. Hier sollte jedoch der Detaillierungsgrad entsprechend dem Untersuchungsziel passend gewählt werden.

Eine Erweiterung der Detaillierungstiefe der Propellermodele um ein Blade-Element-Momentum-Theory Modell [15] wurde im Zuge dieser Arbeit in Betracht gezogen, da insbesondere eine seitliche Anströmung der Propellerebene zu großen Kräften führen kann. Eine Umsetzung erfolgte jedoch bisher nur in Einzelmodellen und noch nicht im Gesamtmodellverbund. Sie wurde bisher wegen der dadurch zu erwartenden Erhöhung der Simulationsdauer und der benötigten Reduktion der Simulationsschrittweite nicht umgesetzt.

Auf einem 3-Freiheitsgradeprüfstand (BILD 1) wurden bereits Versuche durchgeführt, die eine zukünftige Modellvalidierung und Nachparametrierung ermöglichen. Für das System mit vier Propellern konnten aufgrund der Größe auch Versuche zur Validierung im Windkanal vorgenommen werden [16]. Zur schrittweisen Inbetriebnahme des Systems wurde der Albacopter 0.5 zunächst durch ein Kranseil gesichert geflogen (BILD 13). Der Erstflug des Systems konnte bereits erfolgreich vorgenommen werden (BILD 14).



BILD 13. Albacopter® 0.5 im Flug am Kran



BILD 14. Albacopter® 0.5 im Freiflug

## 7. DANKSAGUNG

Die Autoren bedanken sich bei Stephan Myschik für die Durchsicht und Korrektur des Manuskriptes.

## 8. QUELLEN

- [1] J. Jahneke, U. Nolte, M. Henkenjohann, T. Seidenberg, C. Henke, and A. Trächtler, "Development and Implementation of a Modular Interface for a DroneCAN Communication Bus," in *2024 IEEE Aerospace Conference*, Big Sky, MT, USA, 2024, pp. 1–8.
- [2] M. Henkenjohann, U. Nolte, C. Henke, and A. Trächtler, "Novel Cascaded Incremental Nonlinear Dynamic Inversion Controller Approach for a Tiltrotor VTOL," in *2023 International Conference on Unmanned Aircraft Systems (ICUAS)*, Warsaw, Poland, 2023, pp. 1097–1105.
- [3] R. Leitner and S. Myschik, "Design of a Multi-Body 14d Aircraft Model for Ground Handling Evaluation," in *AIAA Modeling and Simulation Technologies Conference*, Toronto, Ontario, Canada, 2010.
- [4] F. Holzapfel, R. Leitner, and G. Sachs, "High Fidelity Landing Gear Modeling for Real-Time Simulation," in *AIAA Modeling and Simulation Technologies Conference and Exhibit*, Keystone, Colorado, 2006.
- [5] R. Leitner and P. Koppitz, "Quick Modeling of Fixed- and Rotary-Wing Aircrafts using a synthetic tree structured Multibody Approach," in *AIAA Modeling and Simulation Technologies Conference*, Minneapolis, Minnesota, 2012.
- [6] R. Leitner, "Coaxial Rotorcraft 18d Multibody Dynamics, The Equations of Motion," in *AIAA Modeling and Simulation Technologies Conference and Exhibit*, Honolulu, Hawaii, 2008.
- [7] J. Pei and C. Roithmayr, "Equations of Motion for a Generic Multibody Tilt-rotor Aircraft," in *AIAA AVIATION 2022 Forum*, Chicago, IL & Virtual, 2022.
- [8] J.-C. Piedboeuf, "Kane's equations or Jourdain's principle?," in *Proceedings of 36th Midwest Symposium on Circuits and Systems*, Detroit, MI, USA, 1993, pp. 1471–1474.
- [9] Y. H. Poy, M. Zarnack, M. Henkenjohann, and U. Nolte, "Aerodynamic Derivatives Identification of a Fixed-Wing UAV using Flight Data," in *AIAA SCITECH 2024 Forum*, Orlando, FL, 2024.
- [10] H. Schlichting and E. Truckenbrodt, *Aerodynamik des Flugzeuges*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001.
- [11] P. Hartmann, "Predictive Flight Path Control for Tilt-Wing Aircraft," RWTH Aachen University, 2017.
- [12] S. E. Mattsson and G. Söderlind, "Index Reduction in Differential-Algebraic Equations Using Dummy Derivatives," *SIAM J. Sci. Comput.*, vol. 14, no. 3, pp. 677–692, 1993, doi: 10.1137/0914043.
- [13] M. Henkenjohann, U. Nolte, F. Sion, C. Henke, and A. Trächtler, "Parameter Tuning Approach for Incremental Nonlinear Dynamic Inversion-Based Flight Controllers," *Actuators*, vol. 13, no. 5, p. 187, 2024, doi: 10.3390/act13050187.
- [14] A. Suti, G. Di Rito, and G. Mattei, "Development and Experimental Validation of Novel Thevenin-Based Hysteretic Models for Li-Po Battery Packs Employed in Fixed-Wing UAVs," *Energies*, vol. 15, no. 23, p. 9249, 2022, doi: 10.3390/en15239249.
- [15] R. L. Rubin and D. Zhao, "New Development of Classical Actuator Disk Model for Propellers at Incidence," *AIAA Journal*, vol. 59, no. 3, pp. 1040–1054, 2021, doi: 10.2514/1.J059734.
- [16] M. Henkenjohann *et al.*, "Dynamic Wind Tunnel Testing of an INDI-Based Flight Controller for a Tiltrotor-VTOL," in *AIAA SCITECH 2025 Forum*, Orlando, FL, 2025.