Decentralized Attitude Stabilization and Formation Control for Aerial Transportation Applications

Joshua Näf
ETH Zürich
Zürich, Switzerland
naefjo@ethz.ch

Franz Bühlmann
ETH Zürich
Zürich, Switzerland
franzbu@ethz.ch

Benjamin Hoffman
ETH Zürich
Zürich, Switzerland
bhoffman@ethz.ch

Robin Ofner
ETH Zürich
Zürich, Switzerland
roofner@ethz.ch

***Abstract*— This paper presents a control algorithm for a multi-aerial manipulator system with the aim of achieving equal load sharing. The system consists of aerial manipulators each equipped with robotic arms allowing them to modify their position during flight while maintaining contact with the object to be transported. The proposed control law ensures stable attitude control of the manipulated object, even in the presence of unknown center of mass positions. The formation of the aerial manipulators is controlled via relative state-based control to avoid agent-agent collisions. This paper provides a mathematical formulation of the problem and presents the control policy which stabilizes the attitude of the carried object. Simulation results further demonstrate the effectiveness of the proposed approach. Lastly, potential improvements and possible directions for further research are also discussed.**

I. INTRODUCTION

Micro Aerial Vehicles (MAVs) are already emerging in industry for visual inspection, mapping and surveillance tasks [1], [2]. A new frontier that presents itself is the integration of MAVs into applications that require manipulation with the environment, so called Aerial Manipulators (AMs) [3]. This problem domain is a rising area of research and has vast applications for single- and multi-MAV systems. Previous work on single-AM systems have already demonstrated several of their applications such as accurate force and torque exertion on their environment [4] and interaction with objects [5]. Further, single-AM systems are already applied in industry for non-destructive testing applications such as wall thickness measurements for steel tanks [6]. On the other hand, multi-AM systems are of interest if large or heavy objects require interaction or transportation, as a single AM's carrying capacity cannot be scaled infinitely. The simple solution, therefore, is to share the load among multiple AMs. Multi-AM systems benefit from increased payload-to-weight ratios as the cumulative thrust of the system increases, as well as from increased flight time as the load on each specific AM decreases with the number of AMs in the system. By using non-rigid connections between the object and the AMs, such as ropes or robotic manipulators, the object can be articulated and its pose can be controlled. This renders AMs increasingly interesting for specialized applications such as collaborative load carrying of heavy payloads on construction sites or removing waste from rivers by the use of a net [7], [8].

Having outlined the increasing relevance of such collaborative systems, the question of control naturally arises. We categorize Multi-AM system control architectures into two distinct classes: centralized and decentralized. An architecture is classified as centralized, if a central planner coordinates the actions of all agents. In other words, full state information about the collaborating agents is needed to formulate a control action for a specific robot. For instance, in [9], the authors propose a nonlinear Model-Predictive Controller (MPC) to control the actions of the involved MAVs while also considering the full pose of a load suspended by cables. In [10], a geometric controller is proposed that takes advantage of model information to control the motion of a payload suspended from cooperating quadrotors by rigid massless links. In [11], four quadrotors are used to transport a suspended payload. All four quadrotors share communication with each other and hence form a rigid formation where relative state based control is used to ensure that the quadrotors hold their formation.

On the other hand, we classify decentralized control architectures as approaches where the connectivity graph between the collaborating robots is sparse, e.g. only neighbouring robots or those within a certain vicinity are able to communicate with each other. A classic example of decentralized control is the leader-follower principle, as demonstrated in [12]. The authors demonstrate collaborative object transportation of a bar by two MAVs, where the designated leader tracks a predefined trajectory and the designated follower uses an admittance control law to guarantee compliance with the leader MAV. In [13], a decentralized MPC is designed for the task of two robots carrying a bar. The model is simplified by assuming that the bar applies the equilibrium force to each MAV and the control actions for both MAVs are computed by two independent MPCs.

A major factor in the flight time of a MAV is its rotor speed, where the required power of a motor scales proportionally with the demanded thrust [14] which negatively affects flight time. Since for multi-MAV systems the total flight time of the system is limited by the lowest flight time of each MAV, if we wish to maximize the total endurance of the system we need to obtain equal load sharing between all MAVs in the system. Of the proposed controllers in the studied literature, none explicitly consider this crucial aspect.

In this work, a control algorithm for a multi-AM system is proposed, where equal load sharing is prioritized. For this approach, it is assumed that each AM is endowed with a sufficiently dexterous robotic manipulator or any similar mechanism to modify its relative position with respect to the center of mass of the object during flight. The proposed control law is capable of stabilizing the angular rate of the manipulated object as well as its attitude, even for an a priori unknown center of mass position of the object. Further, relative state based control is used to control the formation of the AMs and to ensure that agent-agent collisions are avoided. To this end, it is considered that each AM is able to detect its relative distance to its neighbours.

This paper is structured as follows: section II introduces the mathematical formulation of the problem at hand, section III introduces the chosen controller and demonstrates stability properties of the closed-loop system. In section IV the proposed approach is demonstrated in simulation. Finally, section V concludes this paper and gives an overview of potential improvements to the proposed architecture.

II. Problem Formulation

We concern ourselves with the problem of $n \in \mathbb{N}_{\geq 3}$ AMs which are interacting with an arbitrary object for which a rough estimate of its center of mass ${}_I\tilde{\boldsymbol{r}}_{\text{com}}$ is available.

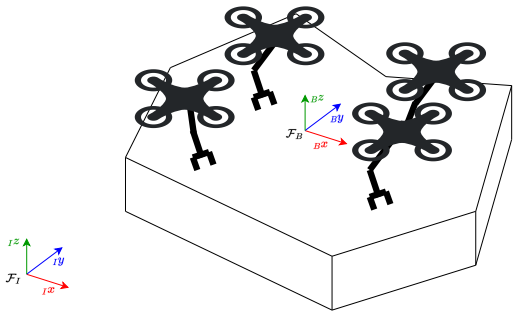

Fig. 1. Problem formulation for $n = 4$ AMs interacting with an arbitrarily shaped object. The manipulators allow the AMs to change their position relative to the object.

While frame $\mathcal{F}_I$ denotes the inertial frame, the body frame $\mathcal{F}_B$ is defined as the body fixed frame centered in the actual center of mass of the object ${}_I\boldsymbol{r}_{\text{com}}$. Furthermore, it is assumed that a stabilizing velocity controller exists for each AM (such as e.g. [15], [16]). Hence, each AM is governed by the following dynamics:

$$ {}_B\dot{\boldsymbol{r}}_i(t) = {}_B\boldsymbol{u}_i(t), \quad i = 1, \dots, n \tag{1} $$

where ${}_B\boldsymbol{r}_i \in \mathbb{R}^3$ denotes the position of AM i expressed in the frame $\mathcal{F}_B$ and ${}_B\boldsymbol{u}_i \in \mathbb{R}^3$ denotes the corresponding velocity control command of AM i in the body frame .

Further, the attitude dynamics of the manipulated object are considered. Let ${}_B\boldsymbol{\omega}_{IB} \in \mathbb{R}^3$ denote the angular rate of the object, ${}_B\boldsymbol{J} \in \mathbb{R}^{3\times 3}$ the inertia tensor of the object and ${}_B\boldsymbol{F}_i \in \mathbb{R}^3$ the force which AM i exerts on said object. It is assumed that the inertia tensor is diagonally dominant and can be approximated well by a diagonal matrix. Also, let $[\boldsymbol{a}]_\times \in \mathbb{R}^{3\times 3}$ denote the skew-symmetric matrix of vector $\boldsymbol{a} \in \mathbb{R}^3$ which is defined as follows

$$ [\boldsymbol{a}]_\times := \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}, \tag{2} $$

and has the property $\boldsymbol{a} \times \boldsymbol{b} = [\boldsymbol{a}]_\times \boldsymbol{b}$.

The attitude dynamics are thus governed by

$$ {}_B\dot{\boldsymbol{\omega}}_{IB} = {}_B\boldsymbol{J}^{-1} \sum_{i=1}^{n} [-{}_B\boldsymbol{F}_i]_\times \, {}_B\boldsymbol{r}_i, \tag{3} $$

where the anticommutativity of the cross product is exploited, ${}_B\boldsymbol{r}_i \times {}_B\boldsymbol{F}_i = -{}_B\boldsymbol{F}_i \times {}_B\boldsymbol{r}_i$, and the skew operator is used to replace the cross product. Since an equal load sharing of the AMs is desired, it is required that all AMs exert the same force on the object and hence let ${}_B\boldsymbol{F}_i = \begin{bmatrix} 0 & 0 & F \end{bmatrix}^\top$, where $F = m_{\text{obj}} g / n$ is chosen such that it compensates for the mass of the object, where m_{obj} is the object's mass and g denotes the gravitational constant. We assume that the aerial manipulator operates in such a fashion that each AM is parallel to the object at all times. Formally, if we denote the angular rate of AM i using ${}_B\boldsymbol{\omega}_i$, then $\boldsymbol{\omega}_i(t) = {}_B\boldsymbol{\omega}_{IB}(t)\ \forall i = \{1, \dots, n\}\ \forall t$.

Finally, let $\boldsymbol{q}_{IB} = \begin{bmatrix} q_w & \boldsymbol{q}_{x,y,z} \end{bmatrix}^\top \in SO(3)$ denote the unit quaternion which describes the rotation from $\mathcal{F}_B$ to $\mathcal{F}_I$. The attitude dynamics are then governed by

$$ \dot{\boldsymbol{q}}_{IB} = \begin{bmatrix} \dot{q}_w \\ \dot{\boldsymbol{q}}_{x,y,z} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} -\boldsymbol{q}_{x,y,z}^\top \, {}_B\boldsymbol{\omega}_{IB} \\ ([\boldsymbol{q}_{x,y,z}]_\times + q_w \mathbb{I}_3) \, {}_B\boldsymbol{\omega}_{IB} \end{bmatrix} \tag{4} $$

Therefore, the total state of the system under analysis is given by

$$ \boldsymbol{x}(t) = \begin{bmatrix} \boldsymbol{q}_{IB} \\ {}_B\boldsymbol{\omega}_{IB}(t) \\ {}_B\boldsymbol{r}_1(t) \\ \vdots \\ {}_B\boldsymbol{r}_n(t) \end{bmatrix} \in \begin{bmatrix} SO(3) \\ \mathbb{R}^{3+3n} \end{bmatrix} \tag{5} $$

III. Controller Design and Analysis

We break the control problem down into three distinct controllers. First, in section III-A we will introduce the attitude rate controller and provide a stability proof. In section III-B, we define the formation control law that regulates the interagent distances. In section III-C, we provide a stability proof for the system with both the attitude rate and the formation controller in the loop. Lastly, in section III-D we introduce the outer loop of the attitude control law. A complete overview of the proposed control architecture can be seen in fig. 2.

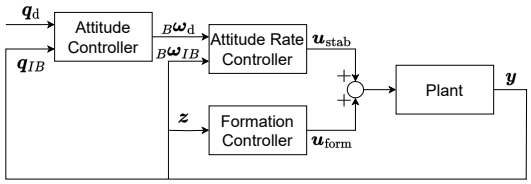

Fig. 2. Block Diagram of the proposed control architecture. Here $\boldsymbol{q}_\mathrm{d}$ refers to the desired attitude, ${}_B\boldsymbol{\omega}_\mathrm{d}$ to the desired attitude rate, $\boldsymbol{u}_\mathrm{stab}$ to the input from the stabilizing controller, $\boldsymbol{u}_\mathrm{form}$ to the input from the formation controller, and $\boldsymbol{z}$ to the relative states.

A. Attitude Rate Controller

By differentiating eq. (3) with respect to time the following linear second order system is obtained:

$$\begin{aligned} {}_B\ddot{\boldsymbol{\omega}}_{IB} &= \boldsymbol{J}^{-1}\left[-\boldsymbol{F}\right]_\times \sum_{i=1}^{n} {}_B\dot{\boldsymbol{r}}_i \\ &\overset{(1)}{=} \boldsymbol{J}^{-1}\left[-\boldsymbol{F}\right]_\times \sum_{i=1}^{n} {}_B\boldsymbol{u}_i \end{aligned} \tag{6}$$

Due to the multiple redundancy in the inputs, we set ${}_B\boldsymbol{u}_i = {}_B\bar{\boldsymbol{u}}/n$ and propose the following control law to stabilize the angular rate

$${}_B\bar{\boldsymbol{u}} = \left[\boldsymbol{F}\right]_\times \left(k_\mathrm{p}({}_B\boldsymbol{\omega}_\mathrm{d} - {}_B\boldsymbol{\omega}_{IB}) + k_\mathrm{d}({}_B\dot{\boldsymbol{\omega}}_\mathrm{d} - {}_B\dot{\boldsymbol{\omega}}_{IB})\right), \tag{7}$$

which only depends on estimates of local quantities and can therefore be implemented in a distributed fashion. Plugging eq. (7) into eq. (6) leads to the following closed-loop dynamics

$$\begin{aligned} {}_B\ddot{\boldsymbol{\omega}}_{IB} = \boldsymbol{J}^{-1}\left[-\boldsymbol{F}\right]_\times \left[\boldsymbol{F}\right]_\times \big(&k_\mathrm{p}({}_B\boldsymbol{\omega}_\mathrm{d} - {}_B\boldsymbol{\omega}_{IB}) \\ &+ k_\mathrm{d}({}_B\dot{\boldsymbol{\omega}}_\mathrm{d} - {}_B\dot{\boldsymbol{\omega}}_{IB})\big). \end{aligned} \tag{8}$$

Assuming ${}_B\ddot{\boldsymbol{\omega}}_\mathrm{d} = 0$, we can reformulate the dynamics in the equivalent linear time-invariant state-space form

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\Delta\boldsymbol{\omega}\\ \Delta\dot{\boldsymbol{\omega}}\end{bmatrix} = \underbrace{\begin{bmatrix}\mathbb{0}_{3\times3} & \mathbb{I}_{3\times3}\\ -k_\mathrm{p}\boldsymbol{M} & -k_\mathrm{d}\boldsymbol{M}\end{bmatrix}}_{:=\boldsymbol{A}_\omega}\begin{bmatrix}\Delta\boldsymbol{\omega}\\ \Delta\dot{\boldsymbol{\omega}}\end{bmatrix} \tag{9}$$

with $\Delta\boldsymbol{\omega} \coloneqq {}_B\boldsymbol{\omega}_\mathrm{d} - {}_B\boldsymbol{\omega}_{IB}$ and $\boldsymbol{M} \coloneqq \boldsymbol{J}^{-1}\left[-\boldsymbol{F}\right]_\times\left[\boldsymbol{F}\right]_\times$.

1) Attitude Rate Stability Analysis: In order to study the stability of the system in eq. (9), Lyapunov stability theory applied to linear systems is used. The following theorem states the relationship between the stability of the system and the condition on the matrix $\boldsymbol{A}_\omega$.

Theorem 1 (Lyapunov Stability for Linear Systems [17]): For an autonomous linear time-invariant system $\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x}$ where $\boldsymbol{x} \in \mathbb{R}^n, \boldsymbol{A} \in \mathbb{R}^{n\times n}$, the following statements are equivalent:

1) The origin $\mathbb{0} \in \mathbb{R}^n$ is an asymptotically stable equilibrium point of the system.
2) All eigenvalues of $\boldsymbol{A}$ are in the open left half plane; $\mathfrak{Re}(\lambda_i(\boldsymbol{A})) < 0\ \forall i = \{1,\dots,n\}$

Since the chosen configuration of AMs is unable to exert a torque around the body z-axis, yaw remains uncontrollable using the proposed control law. Therefore the z-components are removed from the system in eq. (9) for the extent of this analysis. Computing the eigenvalues of the reduced system yields

$$\lambda_{1,2,3,4} = \frac{-F^2k_\mathrm{d}n \pm \sqrt{F^4k_\mathrm{d}^2n^2 - 4F^2j_ik_\mathrm{p}n}}{2j_i},\ i = 1,2, \tag{10}$$

where j_i indicates the entry $({}_BJ)_{i,i}$ of the inertia tensor. If we impose $k_\mathrm{d} > 0$ and further assume that $F, j_i, n > 0$, then $\mathfrak{Re}(\lambda_i) > 0$ can only be achieved if the expression inside the square root is positive, in which case we obtain

$$\begin{aligned} -F^2k_\mathrm{d}n + \sqrt{F^4k_\mathrm{d}^2n^2 - 4F^2j_ik_\mathrm{p}n} &< 0 \\ F^4k_\mathrm{d}^2n^2 - 4F^2j_ik_\mathrm{p}n &< F^4k_\mathrm{d}^2n^2 \\ k_\mathrm{p} &> 0 \end{aligned} \tag{11}$$

Hence, the closed loop system in eq. (9) is asymptotically stable under the proposed control law assuming both control gains are positive. Since the derivation of the control law is independent of the number of robots, the control law could also be applied to a single AM while the other ones are kept steady and asymptotic convergence would still be obtained. This results in a decentralized leader-follower algorithm which is further studied in section IV.

B. Formation Control

In order to avoid agent-agent collisions and to spread the AMs evenly around the center of mass, a formation controller is used which enforces predefined relative distances $\boldsymbol{z}_\mathrm{ref}$ between neighboring AMs. This is achieved by using relative state-based control.

We define the formation to be a regular polygon with n vertices and side length $z_\mathrm{ref} \in \mathbb{R}_+$, where we use the static communication graph $\mathcal{G}$ as depicted in fig. 3, where each AM $i = \{1,\dots,n-1\}$ has a bidirectional communication with its neighbour $i+1$.

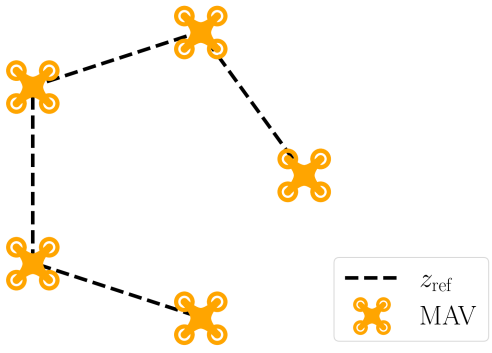

Fig. 3. Proposed communication graph for $n = 5$ AMs. Note that the graph is a spanning tree.

This graph $\mathcal{G}$ is associated with the following incidence

matrix:

$$\boldsymbol{D} = \begin{bmatrix} -1 & 0 & 0 & 0 & \dots & 0 \\ 1 & -1 & 0 & 0 & \dots & 0 \\ 0 & 1 & -1 & 0 & \dots & 0 \\ 0 & 0 & 1 & -1 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \dots & -1 \\ 0 & 0 & 0 & 0 & \dots & 1 \end{bmatrix} \in \mathbb{R}^{n \times n-1} \quad (12)$$

To compute $\boldsymbol{z}_{\text{ref}}$, we can first define the formation by specifying the reference coordinates $\boldsymbol{r}_{\text{ref},i} \in \mathbb{R}^2$ of each AM by equally spacing them on a circle centered in the estimated center of mass ${}_B\tilde{\boldsymbol{r}}_{\text{com}}$. Then, $\boldsymbol{z}_{\text{ref}}$ can be computed using the incidence matrix as

$$\boldsymbol{z}_{\text{ref}} = (\boldsymbol{D}^\top \otimes \mathbb{I}_2)\boldsymbol{r}_{\text{ref}} \in \mathbb{R}^{2(n-1)}, \quad (13)$$

where the binary operator $\otimes : \mathbb{R}^{m\times n}, \mathbb{R}^{p\times q} \rightarrow \mathbb{R}^{mp\times nq}$ denotes the kronecker product between two matrices. Using the incidence matrix $\boldsymbol{D}$ and the desired relative states $\boldsymbol{z}_{\text{ref}}$, we can implement the relative state based control law as proposed in [18]

$$\boldsymbol{u}_{\text{form}}(t) = k_{\text{form}}(\boldsymbol{D} \otimes \mathbb{I}_2)(\boldsymbol{z}_{\text{ref}} - \boldsymbol{z}(t)) \in \mathbb{R}^{2n}, \quad (14)$$

where $\boldsymbol{z}(t)$ denotes the current relative states measured by the AMs and k_{form} denotes a control gain to adjust the rate of convergence to the desired relative state references. The input to each AM is then constructed as

$$\boldsymbol{u}_{i,\text{form}} = \begin{bmatrix} \boldsymbol{u}_{\text{form}}[2i] \\ \boldsymbol{u}_{\text{form}}[2i+1] \\ 0 \end{bmatrix} \quad \forall i = \{1, \dots, n\}, \quad (15)$$

where $\boldsymbol{u}_{\text{form}}[i]$ denotes the i^{th} element of the vector $\boldsymbol{u}_{\text{form}}$.

Defining the error $\boldsymbol{e}(t) = \boldsymbol{z}_{\text{ref}} - \boldsymbol{z}(t)$, we arrive at the following error dynamics

$$\dot{\boldsymbol{e}}(t) = -k_{\text{form}}(\boldsymbol{L}_{\text{edge}} \otimes \mathbb{I}_2)\boldsymbol{e}(t), \quad (16)$$

where the the identity $(\boldsymbol{A} \otimes \boldsymbol{B})(\boldsymbol{C} \otimes \boldsymbol{D}) = (\boldsymbol{AC}) \otimes (\boldsymbol{BD})$ was used to obtain $\boldsymbol{L}_{\text{edge}} = \boldsymbol{D}^\top \boldsymbol{D}$ which describes the edge laplacian of the communication graph $\mathcal{G}$. Since the edge laplacian $\boldsymbol{L}_{\text{edge}}$ is positive definite for a spanning tree, the error converges asymptotically and hence the system is asymptotically stable [18].

C. Low level Controller stability analysis

In this section, we analyze the stability of the system if we apply both control laws as introduced in eqs. (7) and (14). To this end, we investigate the following states:

$$\boldsymbol{x} = \begin{bmatrix} \Delta\boldsymbol{\omega} \\ \Delta\dot{\boldsymbol{\omega}} \\ \boldsymbol{e} \end{bmatrix} \quad (17)$$

For the closed loop system to be stable, we wish to show that said states converge to zero in the closed loop system. Placing eqs. (7) and (14) into eq. (6) and writing the equation in state space form yields

$$\begin{aligned} \dot{\boldsymbol{x}} &= \begin{bmatrix} \Delta\dot{\boldsymbol{\omega}} \\ \boldsymbol{M}(k_{\text{p}}\Delta\boldsymbol{\omega} + k_{\text{d}}\Delta\dot{\boldsymbol{\omega}}) + \boldsymbol{J}^{-1}\left[-\boldsymbol{F}\right]_\times \sum_{i=1}^n \boldsymbol{u}_{i,\text{form}} \\ -k_{\text{form}}(\boldsymbol{L}_{\text{edge}} \otimes \mathbb{I}_2)\boldsymbol{e} \end{bmatrix} \\ &= \begin{bmatrix} \mathbb{0}_{3\times3} & \mathbb{I}_{3\times3} & \mathbb{0}_{3\times2(n-1)} \\ \boldsymbol{M}k_{\text{p}} & \boldsymbol{M}k_{\text{d}} & k_{\text{form}}\boldsymbol{J}^{-1}\left[-\boldsymbol{F}\right]_\times \boldsymbol{\Sigma}(\boldsymbol{D} \otimes \mathbb{I}_2) \\ \mathbb{0}_{2(n-1)\times3} & \mathbb{0}_{2(n-1)\times3} & -k_{\text{form}}(\boldsymbol{L}_{\text{edge}} \otimes \mathbb{I}_2) \end{bmatrix} \boldsymbol{x} \end{aligned} \quad (18)$$

where the matrix $\boldsymbol{\Sigma}$ is chosen such that $\sum_{i=1}^n \boldsymbol{u}_{i,\text{form}} = \boldsymbol{\Sigma}\boldsymbol{u}_{\text{form}}$. Note that we obtain a linear time-invariant system.

By further analysing the previous matrix, we can see that the entry $k_{\text{form}}\boldsymbol{J}^{-1}\left[-\boldsymbol{F}\right]_\times \boldsymbol{\Sigma}(\boldsymbol{D} \otimes \mathbb{I}_2)$ reduces to $\mathbb{0}_{3\times2(n-1)}$ which results in the following system

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \boldsymbol{A}_{\boldsymbol{\omega}} & \mathbb{0}_{6\times2(n-1)} \\ \mathbb{0}_{2(n-1)\times6} & -k_{\text{form}}(\boldsymbol{L}_{\text{edge}} \otimes \mathbb{I}_2) \end{bmatrix} \boldsymbol{x}, \quad (19)$$

which is block diagonal and hence represents two decoupled linear time-invariant systems. Since for a block diagonal matrix $\boldsymbol{A}$ with block matrices $\boldsymbol{A}_1, \dots, \boldsymbol{A}_k$ it holds that $\det(\boldsymbol{A} - \boldsymbol{\lambda I}) = \det(\boldsymbol{A}_1 - \boldsymbol{\lambda I}) \det(\boldsymbol{A}_2 - \boldsymbol{\lambda I}) \dots \det(\boldsymbol{A}_k - \boldsymbol{\lambda I})$, the set of eigenvalues of $\boldsymbol{A}$ are the union of all the sets of eigenvalues of the block matrices: $\{\lambda(\boldsymbol{A})\} = \cup_{i=1}^k \{\lambda(\boldsymbol{A}_i)\}$. Therefore, using the insights gained from sections III-A and III-B, we can infer that the closed-loop system is asymptotically stable if we apply the superposition of the control inputs from eqs. (7) and (14).

D. Attitude Controller

In order to control the attitude of the system, we make use of the proportional controller proposed in [19].

$${}_B\boldsymbol{\omega}_{\text{d}} = k_{\text{q}} \operatorname{sgn}(q_{\text{e},w})\boldsymbol{q}_{\text{e},x,y,z}, \quad (20)$$

where $\boldsymbol{q}_{\text{e}} = \boldsymbol{q}_{IB}^{-1}\boldsymbol{q}_{\text{d}}$ is the error quaternion between the desired attitude $\boldsymbol{q}_{\text{d}}$ and the current attitude $\boldsymbol{q}_{IB}$, k_{q} a control gain and $\operatorname{sgn}(\cdot)$ denotes the sign function as defined below:

$$\operatorname{sgn}(q_{e,w}) = \begin{cases} 1, & q_{e,w} \geq 0 \\ -1, & q_{e,w} < 0 \end{cases} \quad (21)$$

The sign function is necessary since unit quaternions represent a double coverage of $SO(3)$. In other words, $-\boldsymbol{q}$ and $\boldsymbol{q}$ represent the same rotation and hence should produce the same control output as otherwise controller unwinding occurs where the controller rotates a full rotation in the wrong direction, i.e. takes the long way around.

Assuming that the inner loop is significantly faster than the outer loop such that time-scale separation occurs, i.e. ${}_B\boldsymbol{\omega}_{IB} \approx {}_B\boldsymbol{\omega}_{\text{d}}$, it is shown in [19] that the attitude controller is globally asymptotically stable.

IV. Results

The proposed controller has been tested and validated in a numerical simulation. First, in section IV-A we will show the behavior of the controller if all AMs execute the attitude stabilizing control law, subsequently, in section IV-B analyze the behavior of the multi-agent system if only a select few

act as leaders and execute the attitude controller while the others act as followers and solely execute the shape based control policy. The simulation is initialized with n robots which are spaced around an estimate of the center of mass ${}_B\tilde{\boldsymbol{r}}_{\text{com}}$ which, for simplicity, is expressed in the body frame $\mathcal{F}_B$. The initial position can be expressed as

$$\begin{aligned} {}_B\boldsymbol{r}_i(0) &= r\begin{bmatrix}\cos\frac{2\pi i}{n} & \sin\frac{2\pi i}{n}\end{bmatrix} + {}_B\tilde{\boldsymbol{r}}_{\text{com}}, \\ r &\sim \text{Unif}\left(1/2, 3/2\right). \end{aligned} \tag{22}$$

Unless stated otherwise the parameters as shown in table I were used.

Parameter	Value	Description
n	8	Number of AMs
k_{p}	10	Proportional gain for the attitude rate controller
k_{d}	5	Derivative gain for the attitude rate controller
k_{q}	1	Proportional gain for the attitude controller
k_{form}	5	Proportional gain for the formation controller
${}_B\tilde{\boldsymbol{r}}_{\text{com}}$	$[-1.5\ m\ 0.5\ m]^\top$	Initial offset from the center of mass
$\boldsymbol{q}_{\text{d}}$	$[1\ 0\ 0\ 0]^\top$	Desired attitude unit quaternion
$\boldsymbol{\omega}(0)$	$\mathbb{0}_{3\times1}$	Initial angular velocity
$\dot{\boldsymbol{\omega}}(0)$	$\mathbb{0}_{3\times1}$	Initial angular acceleration
z_{ref}	4	Reference interagent distance

TABLE I

NOMINAL SIMULATION PARAMETER VALUES

A. Nominal Controller

We initialize the system according to eq. (22) and table I and let the system evolve to obtain the state trajectories of each AM in reference to one another as well as to the load object as shown in fig. 5. The figure delivers a visual confirmation of the proposed control approach, showing both convergence towards the desired state while maintaining the desired formation.

Figure 4 shows the state and error evolutions plotted against time. The first plot of fig. 4 depicts the attitude described by quaternions. We can observe that the proposed control algorithm is capable of driving the system to the desired attitude $\boldsymbol{q}_{\text{d}}$, exemplified by the attitude of the system converging after 5 seconds with only minimal overshoot present. This simulated behaviour confirms the conclusion made in section III-D in reference to the attitude controller's stability. The implications of not fulfilling the assumptions necessary for global asymptotical stability, i.e. when ${}_B\boldsymbol{\omega}_{IB} \not\approx {}_B\boldsymbol{\omega}_{\text{d}}$, are considered in section IV-C.

The Angular Velocity Evolution can be observed in the second plot of fig. 4. An initial overshoot in ${}_B\omega_y$ with a magnitude of 0.3 rad/s is present due to the relatively poor initial guess of ${}_B\tilde{\boldsymbol{r}}_{\text{com}}$ which results in a torque around the body y axis. However, we can observe that the angular velocity converges to the desired setpoint produced by the attitude controller with less than 10% relative error in approximately 1.75 s. As stated in Section III-A, no torque is present in the body z-axis, which is further evident from the graph; ${}_B\omega_z$ and ${}_B\omega_{\text{d},z}$ remain at 0 throughout the simulation. After a 5-second settling time, the object's intended orientation is reached, and the angular velocities return to zero. Again, the asymptotic stability concluded in section III-A for the attitude rate controller is thus confirmed in simulation.

The third plot in fig. 4 depicts the error evolution over time for all seven relative state errors. The error calculation is based on the error metric $\boldsymbol{e}(t) = \boldsymbol{z}_{\text{ref}} - \boldsymbol{z}(t)$ used for formation control and defined in section III-B. As the original placement of the AMs defines the initial error values, these can vary between simulations and are not further decisive to an evaluation of the control approach. The convergence of all relative state errors however is highly relevant. With all errors converging to zero within approximately 0.75 s, these results demonstrate the asymptotic stability of the proposed formation controller. All AMs rapidly converge and remain in the desired formation which is essential to the collaborative load carrying application proposed.

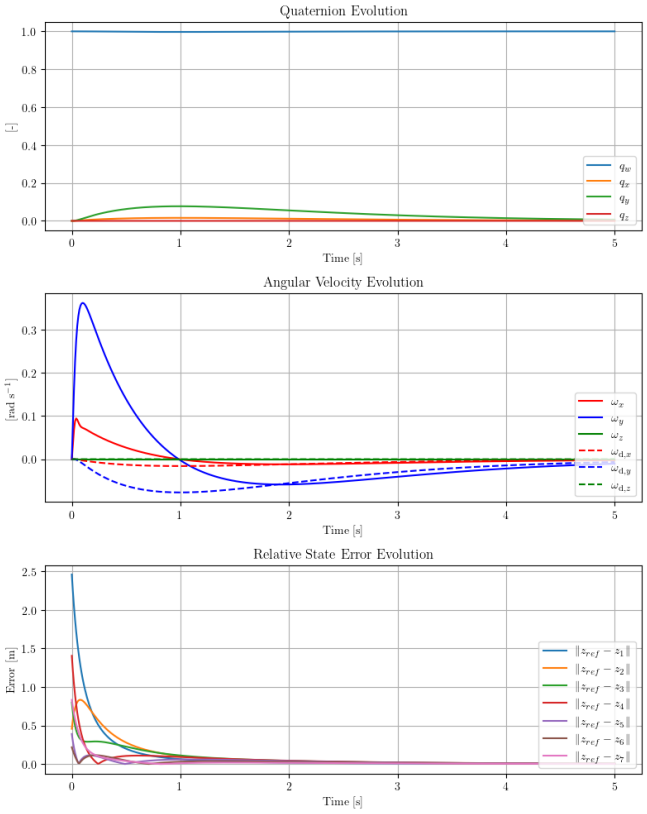

Fig. 4. Nominal state evolution for the attitude, angular velocity, and relative state error in the body frame $\mathcal{F}_B$. The relative error metric, as well as the attitude and angular velocity values, converge.

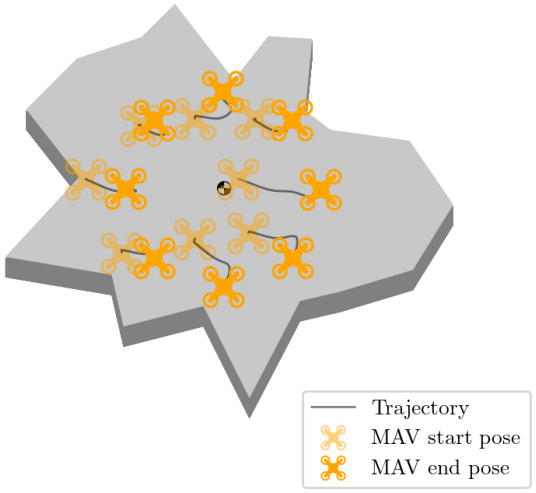

Fig. 5. Nominal state trajectory in context to the load object for $n = 8$ AMs. The AMs clearly converge to the desired positions.

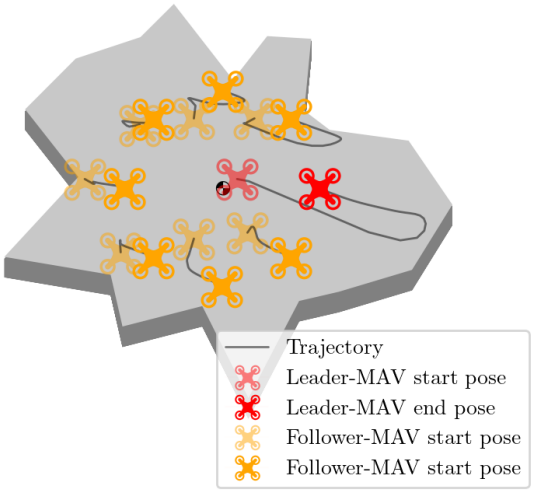

Fig. 6. Leader-follower state trajectory in context to the load object for $n = 8$ AMs. Again, the AMs clearly converge to the desired positions. The individual AM trajectories however are slightly altered.

B. Leader-Follower

In a next step, the decentralized leader-follower control approach referenced in section III-A.1 is simulated and evaluated. In the leader-follower algorithm, only the leader AM executes the attitude stabilizing control law and the follower AMs execute the formation control law, which is compliant with respect to the leader. To obtain the same attitude and attitude rate evolution as in the nominal case, the proportional and derivative gains of the attitude rate controller k_p and k_d are both multiplied by the number of AMs n resulting in $k_\mathrm{p} = 80$ and $k_\mathrm{d} = 40$ for $n = 8$. One can verify using eqs. (6) and (7) that the same closed loop atttiude system response is obtained. Figure 7 depicts the state evolution for the leader-follower control architecture using one leader and seven followers. Further, fig. 6 shows the trajectories of the AMs in reference to each other and the load object.

If the simulation results are compared to those of the nominal controller, the closed-loop covergence congruency of the approaches becomes evident. Both the attitude $\boldsymbol{q}_{IB}$ and the angular velocity ${}_B\boldsymbol{\omega}_{IB}$ evolve identically. As the convergence of the relative state errors is decoupled from the attitude controller, all relative state errors also converge to zero. This not only further confirms the proposed individual controllers, but most importantly validates the implemented leader-follower architecture. While theorized in section III-A, this simulation demonstrates the effectiveness of a decentralized control approach for transportation applications using AMs.

C. Failure Case

To simulate a failure case, let us again study the control architecture shown in fig. 2. While the attitude controller functions well given ${}_B\boldsymbol{\omega}_{IB} \approx {}_B\boldsymbol{\omega}_\mathrm{d}$ as discussed in section III-D, this is only the case if the crossover frequency of the inner loop is sufficiently high. Failing to comply with this condition can be achieved by decreasing the inner loop controller gain k_p, making the inner loop converge slower to the setpoint, or correspondingly by increasing the outer loop gain k_q, making the attitude controller react faster to changes in attitude. In the simulated case, the inner loop gain k_p is decreased to $k_\mathrm{p} = 1$ while k_q is kept constant. The inner loop derivative gain k_d is also decreased to $k_\mathrm{d} = 0.1$ in order to effectively demonstrate the failure case. The resulting instability is apparent when fig. 8 is regarded. Since the assumption of accurate angular velocity tracking is not valid anymore, the lack of time-scale separation leads to oscillations in both the attitude and correspondingly in the angular velocity.

Considering the second plot depicting the angular velocity evolution, it is evident that the low proportional gain makes the tracking of the desired angular velocity ${}_B\boldsymbol{\omega}_\mathrm{d}$ impossible. This is then in turn reflected in the attitude evolution, which is unable to converge due to large overshoot and unreliable angular rate control. This demonstrated failure case underlines the necessity of taking the control architecture and its underlying assumptions into consideration when tuning the respective gains and applying the controller.

V. Conclusion

To summarize, this paper proposes a promising decentralized attitude stabilization and formation control design for collaborative load carrying using AMs with focus on equal load sharing. The suggested control algorithm is composed of attitude rate control, formation control and attitude control, with each individual controller being introduced and

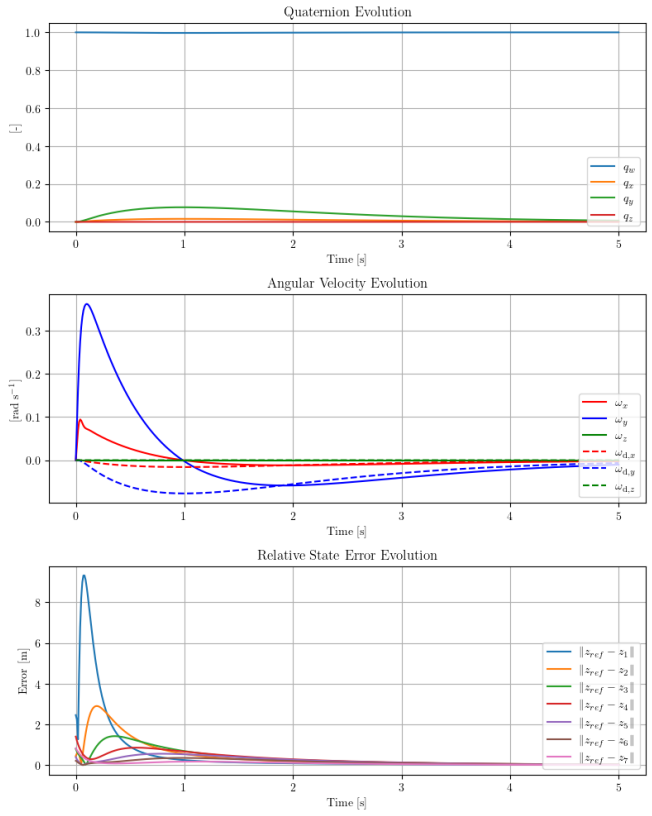

Fig. 7. Leader-follower state evolution for the attitude, angular velocity, and relative state error in the body frame $\mathcal{F}_B$. Again the relative error metric, as well as the attitude and angular velocity values converge.

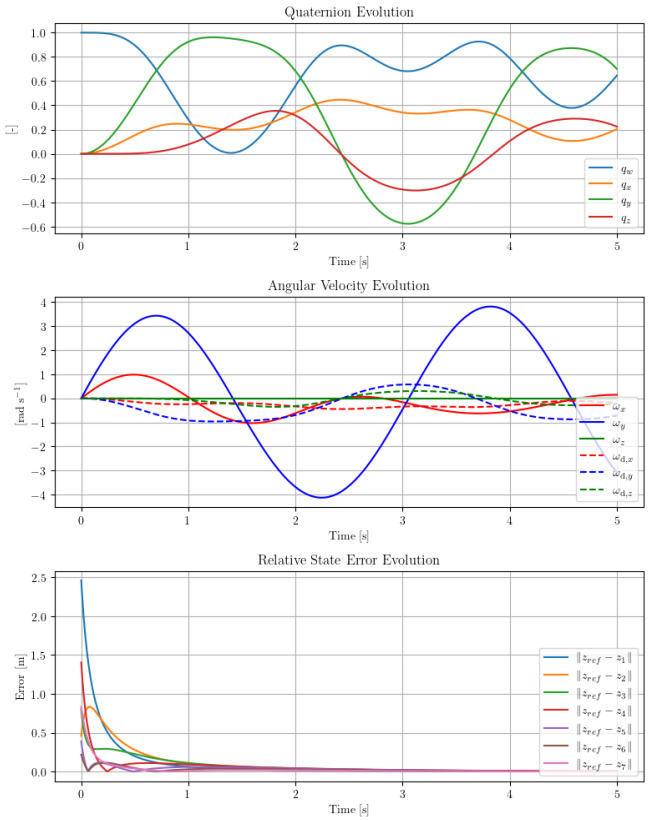

Fig. 8. If the assumption of accurate angular velocity tracking is not applicable, we end up with an unstable state evolution. Again, attitude quantities are expressed in $\mathcal{F}_B$.

a stability proof provided. Further, the provided approach was analyzed and evaluated in simulation. The convergence of attitude, angular velocity and relative state error confirm the theoretical stability results for both the nominal case, as well as when a leader-follower approach is applied. Further, achieving identical simulation results for attitude convergence and angular velocity convergence in the nominal and leader-follower case validate the proposed application of a decentralized control approach. Lastly, the stability of the attitude controller was discussed when the assumption of a reliable inner loop without time-scale separation does not hold. While the stability of the attitude controller was provided both in theory and validated in simulation, a failure case emerges when the complementary controllers do not function in harmony. By decreasing the inner loop gain, the simulation demonstrated the emerging instability, with both the angular velocity unable to track the setpoint and the attitude thus unable to converge, leading to oscillations. Generally however, the control approach was successfully demonstrated and validated, offering a promising approach for the proposed application of collaborative load sharing using AMs.

A possible extension of the work done in this paper is the augmentation of the control approach to stabilize the load object's position as well. This would further increase the applicability of AMs to such use cases, more delicate objects could be carried and more precise transportation could be conducted. Further, applying distributed optimization to minimize F, the force exertion necessary to carry the load, objects of unknown mass could also be transported. To this end, the problem could be cast into a decentralized MPC formulation with consensus constraints on the mass of the object and appropriate local cost functions. This would allow for broad and instantaneous applicability of the system. Alternatively, a consensus algorithm could be applied initially to estimate the weight of the object in flight. A crucial assumption in the control design is that each AM has a sufficiently dexterous robotic manipulator to change its relative state on the object. This assumption is very relaxing as any robotic manipulator has a limited workspace and can not arbitrarily reach any point in the work space. Thus, the controller should be extended to take these limitations into account. Lastly, by relaxing the condition that the AMs have the same angular velocity as the carried object and instead relying on geometric properties of the system, the complexity of the applied AMs could be reduced by using ropes as attachments to the object instead of manipulators.

REFERENCES

[1] "Flyability's Confined Space Drone Makes Inspections Safer & Cheaper." [Online]. Available: https://www.flyability.com

[2] "Mapping drone for fast and accurate survey data every time." [Online]. Available: https://wingtra.com/

[3] M. Rubio, J. Näf, F. Bühlmann, P. Brigger, M. Hüsser, M. Inauen, N. Ospelt, D. Gisler, M. Tognon, and R. Siegwart, "Design of prismav: An omnidirectional aerial manipulator based on a 3-puu parallel mechanism," in *2023 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2023, pp. 608–615.

[4] K. Bodie, M. Brunner, M. Pantic, S. Walser, P. Pfändler, U. Angst, R. Siegwart, and J. Nieto, "Active Interaction Force Control for Contact-Based Inspection with a Fully Actuated Aerial Vehicle," *IEEE Transactions on Robotics*, vol. 37, no. 3, pp. 709–722, Jun. 2021, arXiv:2003.09516 [cs]. [Online]. Available: http://arxiv.org/abs/2003.09516

[5] S. Shimahara, S. Leewiwatwong, R. Ladig, and K. Shimonomura, "Aerial torsional manipulation employing multi-rotor flying robot," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2016, pp. 1595–1600, iSSN: 2153-0866.

[6] "Voliro: Contact-based inspection drone for NDT and wind turbine LPS." [Online]. Available: https://voliro.com

[7] A. Mohiuddin, T. Tarek, Y. Zweiri, and D. Gan, "A Survey of Single and Multi-UAV Aerial Manipulation," *Unmanned Systems*, vol. 08, no. 02, pp. 119–147, Apr. 2020. [Online]. Available: https://www.worldscientific.com/doi/abs/10.1142/S2301385020500089

[8] C. Gabellieri and A. Franchi, "Differential Flatness and Manipulation of Elasto-flexible Cables Carried by Aerial Robots in a Possibly Viscous Environment," in *2023 International Conference on Unmanned Aircraft Systems (ICUAS)*. Warsaw, Poland: IEEE, Jun. 2023, pp. 963–968. [Online]. Available: https://ieeexplore.ieee.org/document/10156297/

[9] S. Sun and A. Franchi, "Nonlinear MPC for Full-Pose Manipulation of a Cable-Suspended Load using Multiple UAVs," Apr. 2023, arXiv:2301.08545 [cs]. [Online]. Available: http://arxiv.org/abs/2301.08545

[10] T. Lee, K. Sreenath, and V. Kumar, "Geometric control of cooperating multiple quadrotor UAVs with a suspended payload," in *52nd IEEE Conference on Decision and Control*, Dec. 2013, pp. 5510–5515, iSSN: 0191-2216.

[11] H. G. de Marina and E. Smeur, "Flexible collaborative transportation by a team of rotorcraft," Feb. 2019, arXiv:1902.00279 [cs]. [Online]. Available: http://arxiv.org/abs/1902.00279

[12] A. Tagliabue, M. Kamel, S. Verling, R. Siegwart, and J. Nieto, "Collaborative Object Transportation Using MAVs via Passive Force Control," Dec. 2016, arXiv:1612.04915 [cs] version: 1. [Online]. Available: http://arxiv.org/abs/1612.04915

[13] R. C. Sundin, P. Roque, and D. V. Dimarogonas, "Decentralized Model Predictive Control for Equilibrium-based Collaborative UAV Bar Transportation," in *2022 International Conference on Robotics and Automation (ICRA)*, May 2022, pp. 4915–4921.

[14] M. Biczyski, R. Sehab, J. F. Whidborne, G. Krebs, and P. Luk, "Multirotor Sizing Methodology with Flight Time Estimation," *Journal of Advanced Transportation*, vol. 2020, pp. 1–14, Jan. 2020. [Online]. Available: https://www.hindawi.com/journals/jat/2020/9689604/

[15] "Controller Diagrams | PX4 User Guide." [Online]. Available: https://docs.px4.io/main/en/flight_stack/controller_diagrams.html

[16] V. S. Akkinapalli and F. Holzapfel, "Incremental Dynamic Inversion based Velocity Tracking Controller for a Multicopter System," in *2018 AIAA Guidance, Navigation, and Control Conference*. Kissimmee, Florida: American Institute of Aeronautics and Astronautics, Jan. 2018. [Online]. Available: https://arc.aiaa.org/doi/10.2514/6.2018-1345

[17] M. Dahleh, M. A. Dahleh, and G. Verghese, *Lectures on Dynamic Systems and Control*. MIT OpenCourseWare, 2011. [Online]. Available: https://ocw.mit.edu/courses/6-241j-dynamic-systems-and-control-spring-2011/996025f6db0d90b00f11c44fc49b85f9_MIT6_241JS11_textbook.pdf

[18] M. Mesbahi and M. Egerstedt, *Graph theoretic methods in multiagent networks*, ser. Princeton series in applied mathematics. Princeton: Princeton University Press, 2010, oCLC: ocn466341412.

[19] D. Brescianini, M. Hehn, and R. D'Andrea, "Nonlinear Quadrocopter Attitude Control: Technical Report," ETH Zurich, Tech. Rep., 2013, medium: application/pdf,Online-Ressource. [Online]. Available: http://hdl.handle.net/20.500.11850/154099