Computational Mechanics
https://doi.org/10.1007/s00466-026-02758-x

ORIGINAL PAPER

Modular Voxel-DEM based design and analysis of quadcopter drones

T. I. Zohdi[1]

Received: 15 October 2025 / Revised: 8 January 2026 / Accepted: 22 January 2026

Abstract
This work develops a flexible and modular Discrete Element Method paradigm for rapid quadcopter drone analysis and simulation. The main components are (1) modular assembly of structural components and (2) flexible feature addition and deletion and dynamic drone flight performance evaluation. As a model problem, we illustrate the concepts above for the design and analysis of a quadcopter and its subsequent dynamic flight performance in an extreme environment. The outline of the work is (a) modular generation of voxel-based discrete elements, (b) preflight stress testing using voxel-based discrete elements, (c) structural materials for drones, (d) voxel-based discrete elements for dynamic drone flight evaluation and (e) extensions to fully finite-deformation structural flight dynamics. Numerical examples are provided to illustrate the overall approach and its straightforward implementation.

Keywords Drones · Design · Discrete elements

1 Introduction: drones in an evolving society

Over the last 20 years, societal interest in drones has grown dramatically and they have become important industry tools. In particular, quadcopters, which are ideal for detailed and precise "stop and go" operations, have become a mainstay of many drone-based systems. Quadcopter drones are advantageous due to their simplicity, excellent stability, ease of control, maneuverability, and relatively low cost. Their design with four rotors makes them easier to manufacture, maintain, and repair compared to traditional helicopters, which contributes to their affordability. Furthermore, their stability makes them ideal for applications like aerial photography and videography. Specifically, the advantages that quadcopters offer are:

- **Maintenance and transportation:** Quadcopters have fewer moving parts than helicopters, making them easier to understand, repair, and maintain. Quadcopters are lightweight and often foldable, making them easy to transport and deploy.
- **Stability and ease of control:** Quadcopters that utilize advancements in electric motors and power semiconductors are easier to control, especially with the aid of electronic stabilization systems. The four-rotor design provides excellent hovering capabilities, crucial for tasks like aerial photography and videography.
- **Vertical Takeoff and Landing (VTOL):** Quadcopters can take off and land vertically without needing a runway, making them versatile in various environments. They also allow for rapid changes in rotor speed allow for agile movements, enabling them to navigate tight spaces and perform acrobatic maneuvers.
- **Cost-effectiveness:** Quadcopters are more affordable than other drone configurations, such as helicopters, making them accessible to a wide range of users.

For the most recent developments in quadcopter dynamics and control, we refer the readers to Mueller [1], to [2–27] for a wide cross-section of popular and technical literature and Zohdi [28] for an analysis of multi-drone systems.

A key feature of quadcopters is the ability to easily add cameras, sensors, etc. to the chassis. For example many drones now carry a variety of multispectral and LiDAR (Light Detection And Ranging) type cameras, acoustical sensors and associated signal processing tools. In particular, LiDAR has become quite popular and typically uses light in the high-frequency ultraviolet, visible and near infrared spectrum (Ring [29], Cracknell and Hayes [30], Goyer and Watson [31], Medina et al [32] and Trickey et

✉ T. I. Zohdi
zohdi@berkeley.edu

[1] Department of Mechanical Engineering, University of California, Berkeley, CA 94720-1740, USA

Published online: 10 March 2026

al [33]). LiDAR is classified as a "time-of-flight" type technology, utilizing a pulse of light and the time of travel to determine the relative distance of an object. Over the last 20 years, these devices have steadily improved and have become quite lightweight [34–40]. Furthermore, due to the steady increase in inexpensive drones and camera technology, there are a wide variety of applications for drones, such as infrastructural mapping, wildlife monitoring, firefighting, military engagement, aerial photography, remote exploration, archaeology, surveillance, search and rescue operations, agriculture, construction, etc. The number of uses is constantly growing and is already rather immense. These applications may demand (a) complete remote control by humans, (b) autopilot assistance or (c) completely independent autonomy, which include features such as (1) GPS waypoint navigation (2) pre-programmed routes (3) preprogrammed delivery (4) automated take-off and landing (5) failsafe landing and (6) return to home. Furthermore, depending on their intended use, flight duration, altitude, range, etc., their power sources could be any of the following (1) fuel-power sources, which are suitable for high-altitudes and long range operations, for example for military operations, (2) battery-power sources, which are for quiet applications in complex environments, with limited flight times, which make them ideal for agricultural facilities, industrial facilities and cities, (3) hybrid electric-fuel power sources, which are suitable for dual use operations where fuel and battery powered operations are needed and (4) solar-power sources, for extremely long duration and high-altitude operations that involve monitoring large areas. *Thus, in order to tailor a quadcopter for is specific intended use, there is a need for simulation based design and analysis, before one gets to a prototype building stage.* According, the objective of the present work is to develop a flexible modular Discrete Element Method (DEM) paradigm for rapid quadcopter drone analysis and simulation. The main components are (1) modular assembly of structural components and (2) flexible feature addition and deletion and dynamic drone flight performance evaluation. As a model problem, we illustrate the concepts above for the design of a quadcopter and its subsequent dynamic flight performance in an extreme environment. The outline is:

- Modular generation of voxel-based discrete elements
- Preflight stress testing using voxel-based discrete elements
- Structural materials for drones
- Voxel-based discrete elements for dynamic drone flight evaluation
- Extensions to fully finite-deformation structural flight dynamics

2 Modular generation for voxel-based discrete elements

To generate a DEM representation of a quadcopter body, we insert a design envelope within a grid (with resolution specified by the analyst) and determine the intersection in order to generate the active voxel grid units (Figures 1-4). For example, an easy to parametrize, modular, envelope is given by a generalized ellipsoidal equation (Zohdi [28])

$$|\frac{x_1 - x_{1o}}{R_1}|^{p_1} + |\frac{x_2 - x_{2o}}{R_3}|^{p_2} + |\frac{x_3 - x_{3o}}{R_3}|^{p_3} \leq 1 \tag{2.1}$$

where (x_1, x_2, x_3) are the coordinates throughout the domain, (x_{1o}, x_{2o}, x_{3o}) are the coordinates of center of the chassis envelope and (R_1, R_2, R_3) are the generalized radii and (p_1, p_2, p_3) are exponents of the generalized ellipsoid. This generates a set of active voxels within the domain (Figures 1-4). Where there is an intersection, a voxel is considered "active". For exponent values of (p_1, p_2, p_3) equal to two, we generate an ellipsoid, for values less than one we generate involute (nonconvex shapes, Figures 1-4), and for exponent values of (p_1, p_2, p_3) greater than two, we generate a box-like shapes. By computing the union of the active voxel grid sets of multiple intersecting objects, we can created numerous drone designs (Figures 1-4):

$$\Omega^{overall} = \bigcup_{i=1}^{N} \Omega_i = \Omega_1 \cup \Omega_2 \cup \Omega_3 \cup ...\Omega_N, \tag{2.2}$$

and well as subtraction of domains, shown in Figure 3. Any voxel that is in the union is considered "active". *The active voxels will be the initial locations of the Discrete Elements.* Figures 1-4 illustrate the generation of discrete elements, front and back shown. For illustration purposes, the back has a plate added for a possible camera mount. Also shown are various chassis envelopes: (a) $(p_1, p_2, p_3) = (0.5, 0.5, 0.5)$ and (b) $(p_1, p_2, p_3) = (1.0, 1.0, 1.0)$ with "lightweighting" holes punched out for weight reduction. Additional lightweighting of the structure is extremely easy to analyze by simply deleting voxels.

Remark 1 Such an analysis has great utility for preflight stress analysis, which is discussed next. It is important to make the connection between voxels and discrete elements in the upcoming framework. The voxels are used for on-flight stress analysis only, and the discrete elements (constructed over the voxels' grid in a bonded-DEM approach, allowing the representation of arbitrary shapes, such as drone shapes through a particle cluster) for dynamic flight simulation only. In the upcoming analysis, discrete elements will be spheres, using a bonded DEM model (bonded neighbors). Please provide more details at this stage of the manuscript (see upcoming Figure 8).

- Modular generation of voxel-based discrete elements
- Preflight stress testing using voxel-based discrete elements
- Structural materials for drones
- Voxel-based discrete elements for dynamic drone flight evaluation
- Extensions to fully finite-deformation structural flight dynamics

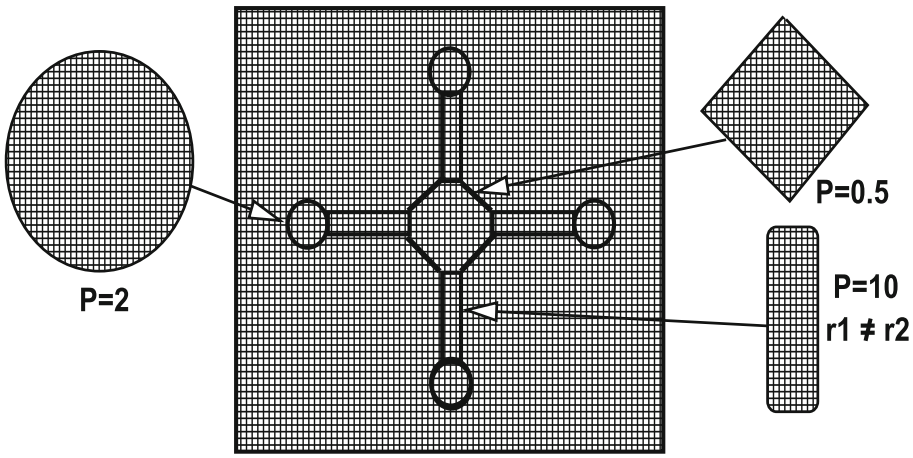

Fig. 1 Chassis building blocks that are generated by combining the basic $F(x_1, x_2, x_3) = 0$ Equation 2.1 with the union of overlapping $\Omega^{overall} = \Omega_1 \cup \Omega_2 \cup \Omega_3 \cup ... \Omega_N$ subdomains

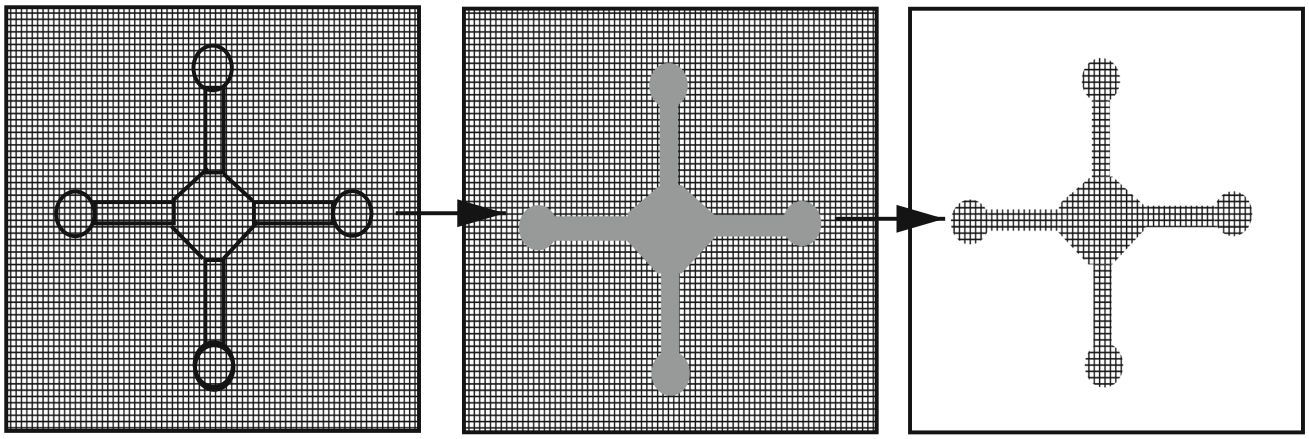

Fig. 2 A sequence of steps that lead to a voxel-DEM representation of a structure

3 Preflight stress testing using voxel-based discrete elements

Although our main objective is to perform a rapid flight analysis, it is important to perform a preflight stress analysis. Because the discrete elements are already in place in a grid-like form (Figures 1-4), it makes little sense to employ the usual Finite Element machinery: topologically conforming meshing, mapping, volume integration, stiffness matrix generation, etc. The voxel-based DEM can then be used to do an initial stress analysis of for a drone. We consider a balance of linear momentum governed by

$$\nabla_x \cdot \boldsymbol{\sigma} + \boldsymbol{f} = \rho \frac{\partial \boldsymbol{v}}{\partial t} = \rho \frac{\partial^2 \boldsymbol{u}}{\partial t^2}, \tag{3.1}$$

in (design) regimes where *infinitesimal deformations are appropriate*, where $\boldsymbol{\sigma}$ is the Cauchy stress, $\boldsymbol{f}$ are body forces, ρ is the material density and $\boldsymbol{u}$ is the displacement. Consistent with the infinitesimal deformation approximation we write $\nabla_x \approx \nabla_X$ and $\frac{d()}{dt} \approx \frac{\partial()}{\partial t}|_X$, where $\boldsymbol{X}$ are the referential coor-

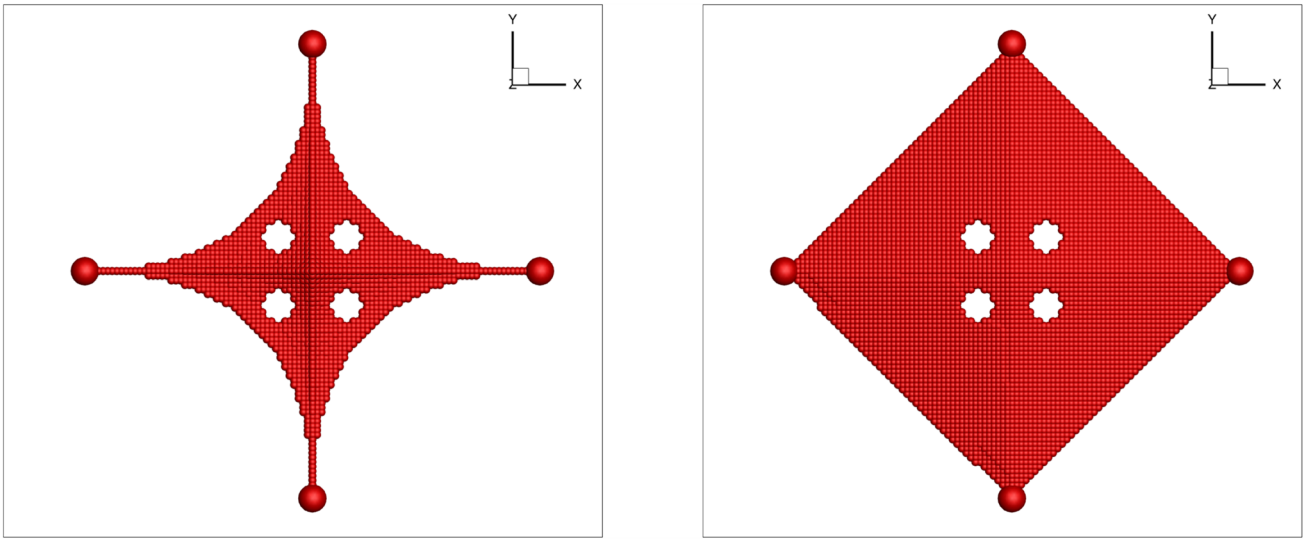

Fig. 3 Generating a component of the drone chassis (the centerpiece) with discrete elements. This achieved by checking for the intersection of the interior of the hull envelope equation above with a box of voxels ($\pm R_1, \pm R_2, \pm R_3$), Figs. 1 and 2). As an example, a generalized ellipsoidal equation (Equation 2.1) is used, where for exponent values of (p_1, p_2, p_3) equal to two, we generate an ellipsoid, for values less than one we generate involute (nonconvex shapes), and for exponent values of (p_1, p_2, p_3) greater than two, we generate box-like shapes. Shown are chassis envelopes ((p_1, p_2, p_3) = (0.5, 0.5, 0.5) and bottom right: (p_1, p_2, p_3) = (1.0, 1.0, 1.0) with "lightweighting" holes punched out for weight reduction. Additional lightweighting of the structure is extremely easy to analyze by simply deleting voxels (based on Zohdi [28])

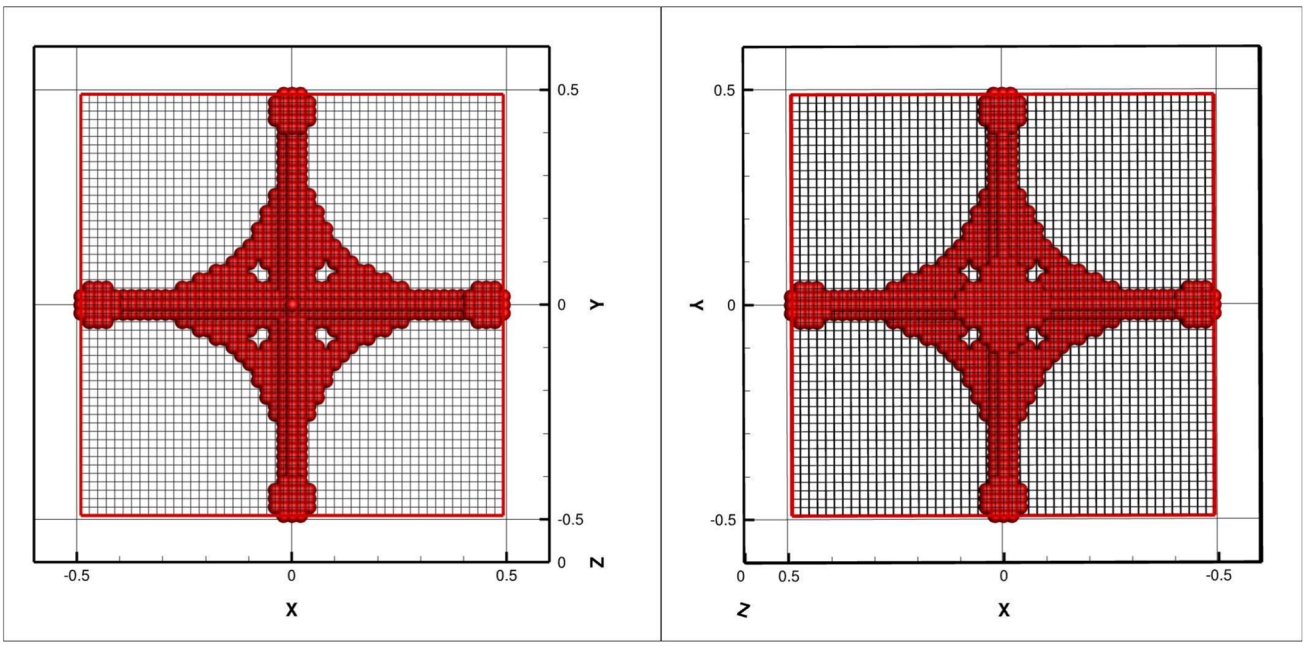

Fig. 4 Generation of active voxels: front and back shown. For illustration purposes, the back has a plate (feature) added for a possible camera mount

dinates, $\boldsymbol{x}$ are the current coordinates ($\boldsymbol{X} \approx \boldsymbol{x}$). We consider a linearly-elastic constitutive law given by

$$\boldsymbol{\sigma} = \mathbb{E} : \boldsymbol{\epsilon}, \tag{3.2}$$

which under infinitesimal deformation framework the balance of linear momentum becomes ($\rho \approx \rho_o$)

$$\nabla_x \cdot (\mathbb{E} : \boldsymbol{\epsilon}) + \boldsymbol{f} = \rho_o \frac{\partial \boldsymbol{v}}{\partial t} = \rho_o \frac{\partial^2 \boldsymbol{u}}{\partial t^2} \tag{3.3}$$

with infinitesimal strains given by $\boldsymbol{\epsilon} = \frac{1}{2}(\nabla_x \boldsymbol{u} + (\nabla_x \boldsymbol{u})^T)$. In many methods that analyze structures, for example 3D printed, the computation of the response of multiple prototype prints are required. The 3D print domain is usually

taken to be a parallel-pipped, but contains a complex substructure, for example a drone. The most frequent application where this occurs is in structural design optimization, where the prototype structures are constantly being changed during the search process. Usually, the Finite Element Method (FEM) is the default method used for such an analysis. The fundamental problem is that the FEM is computationally expensive for structural problems with a regular domain, due to the "ingredients" involved in the usual Finite Element process: meshing, mapping, volume integration, stiffness matrix generation and matrix-based solution methods. This computational expense is even more unwarranted when there is an interest in time-dependent regimes. An alternative family of techniques that is naturally suited for these types of problems are so-called "Digital/Voxel-Image" methods, which convert the material microstructure into voxels (3D "volume pixels"; see Figures 1-4). The problem then becomes "digital" on a regular "voxel-grid", directly manipulating voxel values. Extremely fast methods can then be used to construct derivatives and the solve the system.

The present section develops a flexible and robust solution strategy to resolve the system of equations for the mechanical fields. There are two main components to the computational approach:

- Spatio-temporal discretization of the continuum model,
- Iterative staggering to solve the coupled system, whereby the time-steps are adaptively adjusted to control the error associated with the incomplete resolution of the solution fields.

3.1 Temporal discretization

For the velocity field (infinitesimal deformation formulation) we write

$$\frac{d\boldsymbol{v}}{dt} = \frac{\partial \boldsymbol{v}}{\partial t} = \frac{1}{\rho}\left(\nabla_x \cdot \boldsymbol{\sigma} + \boldsymbol{f}\right) \stackrel{\text{def}}{=} \boldsymbol{\Psi}. \tag{3.4}$$

We discretize for time=$t+\phi\Delta t$, and using a trapezoidal "$\phi-scheme$" ($0 \leq \phi \leq 1$) at a point $\boldsymbol{x} \approx \boldsymbol{X}$

$$\begin{aligned}\frac{\boldsymbol{v}(\boldsymbol{x}, t+\Delta t) - \boldsymbol{v}(\boldsymbol{x}, t)}{\Delta t} &\approx \boldsymbol{\Psi}(\boldsymbol{x}, t+\phi\Delta t)\\ &\approx \phi\boldsymbol{\Psi}(\boldsymbol{x}, t+\Delta t) + (1-\phi)\boldsymbol{\Psi}(\boldsymbol{x}, t).\end{aligned} \tag{3.5}$$

Rearranging, yields

$$\begin{aligned}\boldsymbol{v}(\boldsymbol{x}, t+\Delta t) \approx \boldsymbol{v}(\boldsymbol{x}, t) + \Delta t\,(&\phi\boldsymbol{\Psi}(\boldsymbol{x}, t+\Delta t)\\ &+(1-\phi)\boldsymbol{\Psi}(\boldsymbol{x}, t))\end{aligned} \tag{3.6}$$

where the previously introduced spatial discretization is applied to the terms in $\boldsymbol{\Psi}(\boldsymbol{x}, t)$. Since this is a second-order system, the procedure is then repeated to determine the displacement field $\boldsymbol{u}$

$$\begin{aligned}\boldsymbol{u}(\boldsymbol{x}, t+\Delta t) &= \boldsymbol{u}(\boldsymbol{x}, t) + \boldsymbol{v}(\boldsymbol{x}, t+\phi\Delta t)\Delta t\\ &= \boldsymbol{u}(\boldsymbol{x}, t) + (\phi\boldsymbol{v}(\boldsymbol{x}, t+\Delta t)\\ &\quad +(1-\phi)\boldsymbol{v}(\boldsymbol{x}, t))\,\Delta t,\end{aligned} \tag{3.7}$$

or more explicitly

$$\boldsymbol{u}(\boldsymbol{x}, t+\Delta t) = \boldsymbol{u}(\boldsymbol{x}, t) + \boldsymbol{v}(\boldsymbol{x}, t)\Delta t + \phi(\Delta t)^2\boldsymbol{\Psi}(\boldsymbol{x}, t+\phi\Delta t). \tag{3.8}$$

The term $\boldsymbol{\Psi}(\boldsymbol{x}, t+\phi\Delta t)$ can be handled by the following approximation

$$\boldsymbol{\Psi}(\boldsymbol{x}, t+\phi\Delta t) \approx \phi\boldsymbol{\Psi}(\boldsymbol{x}, t+\Delta t) + (1-\phi)\boldsymbol{\Psi}(\boldsymbol{x}, t). \tag{3.9}$$

Therefore,

$$\begin{aligned}\boldsymbol{u}(\boldsymbol{x}, t+\Delta t) = \boldsymbol{u}(\boldsymbol{x}, t) + \boldsymbol{v}(\boldsymbol{x}, t)\Delta t + \phi(\Delta t)^2\,(&\phi\boldsymbol{\Psi}(\boldsymbol{x}, t\\ &+\Delta t) + (1-\phi)\boldsymbol{\Psi}(\boldsymbol{x}, t))\,.\end{aligned} \tag{3.10}$$

When $\phi = 1$, then this approach can be considered to be a (implicit) Backward Euler scheme, which is very stable (very dissipative) and $\mathcal{O}((\Delta t)^2)$ locally in time, while if $\phi = 0$, the scheme can be considered as a (explicit) Forward Euler scheme, which is conditionally stable and $\mathcal{O}((\Delta t)^2)$ locally in time. If $\phi = 0.5$, then the scheme can be considered as a (implicit) Midpoint scheme, which is marginally stable and $\mathcal{O}((\Delta t)^3)$ locally in time.

3.2 Spatial discretization of the fields

Numerically, the components of the gradient of an arbitrary field variable $\boldsymbol{w}$ are approximated by central finite difference stencils of the basic form, for example $\frac{\partial}{\partial x_1}$

$$\frac{\partial \boldsymbol{w}}{\partial x_1}|_{(x_1,x_2,x_3)} \approx \frac{\boldsymbol{w}(x_1+\Delta x_1, x_2, x_3) - \boldsymbol{w}(x_1-\Delta x_1, x_2, x_3)}{2\Delta x_1} \tag{3.11}$$

for each of the (x_1, x_2, x_3)-directions, in order to form the terms needed. This is a second-order accurate stencil. In order to illustrate the approximation, for a generic second-order scheme, the spatial derivatives for a flux-type term, $\mathcal{F}$, the components $\mathcal{F}_i, i = 1, 2, 3$ are initially approximated with half-step stencils. For example, for $\mathcal{F}_1$ component in the x_1 direction one has

$$\frac{\partial \mathcal{F}_1}{\partial x_1}|_{(x_1,x_2,x_3)} \approx \frac{\mathcal{F}_1(x_1+\frac{\Delta x_1}{2},x_2,x_3)-\mathcal{F}_1(x_1-\frac{\Delta x_1}{2},x_2,x_3)}{\Delta x_1}, \tag{3.12}$$

where generically, for example, with an arbitrary material coefficient $A(\boldsymbol{x})$

$$\left(\underbrace{A\frac{\partial \boldsymbol{w}}{\partial x_1}}_{\mathcal{F}_1}\right)|_{(x_1+\frac{\Delta x_1}{2},x_2,x_3)} \approx A(x_1+\frac{\Delta x_1}{2},x_2,x_3)\underbrace{\frac{\boldsymbol{w}(x_1+\Delta x_1,x_2,x_3)-\boldsymbol{w}(x_1,x_2,x_3)}{\Delta x_1}}_{\frac{\partial \boldsymbol{w}}{\partial x_1}|_{(x_1+\frac{\Delta x_1}{2},x_2,x_3)}} \tag{3.13}$$

and

$$\left(\underbrace{A\frac{\partial \boldsymbol{w}}{\partial x_1}}_{\mathcal{F}_1}\right)|_{(x_1-\frac{\Delta x_1}{2},x_2,x_3)} \approx A(x_1-\frac{\Delta x_1}{2},x_2,x_3)\underbrace{\frac{\boldsymbol{w}(x_1,x_2,x_3)-\boldsymbol{w}(x_1-\Delta x_1,x_2,x_3)}{\Delta x_1}}_{\frac{\partial \boldsymbol{w}}{\partial x_1}|_{(x_1-\frac{\Delta x_1}{2},x_2,x_3)}} \tag{3.14}$$

where

$$A(x_1+\frac{\Delta x_1}{2},x_2,x_3) \approx \frac{1}{2}(A(x_1+\Delta x_1,x_2,x_3)+A(x_1,x_2,x_3)), \tag{3.15}$$

and

$$A(x_1-\frac{\Delta x_1}{2},x_2,x_3) \approx \frac{1}{2}(A(x_1,x_2,x_3)+A(x_1-\Delta x_1,x_2,x_3)). \tag{3.16}$$

These approximations are made for all components and combinations in $A\frac{\partial w}{\partial x_j}$ appearing in the field equations, specifically, for all components and combinations in $\nabla_x \cdot \boldsymbol{\sigma}$ appearing in the field equations:

$$\rho\frac{\partial v_1}{\partial t} = \frac{\partial \sigma_{11}}{\partial x_1}+\frac{\partial \sigma_{12}}{\partial x_2}+\frac{\partial \sigma_{13}}{\partial x_3}+f_1$$
$$\rho\frac{\partial v_2}{\partial t} = \frac{\partial \sigma_{21}}{\partial x_1}+\frac{\partial \sigma_{22}}{\partial x_2}+\frac{\partial \sigma_{23}}{\partial x_3}+f_2$$
$$\rho\frac{\partial v_3}{\partial t} = \frac{\partial \sigma_{31}}{\partial x_1}+\frac{\partial \sigma_{32}}{\partial x_2}+\frac{\partial \sigma_{33}}{\partial x_3}+f_3. \tag{3.17}$$

The mixed derivatives are derived in a similar manner. The mathematical representation of the derivatives can be summarized in the following manner $\frac{\partial}{\partial x_j}$, $j = 1,2,3$, for example for $j = 1$:

1. **VOXEL-GRADIENT:** $A\frac{\partial \boldsymbol{w}}{\partial x_1} \approx A(x_1,x_2,x_3)\frac{\boldsymbol{w}(x_1+\Delta x_1,x_2,x_3)-\boldsymbol{w}(x_1-\Delta x_1,x_2,x_3)}{2\Delta x_1}$
2. **VOXEL-LAPLACIAN:**

$$\begin{aligned}&\frac{\partial}{\partial x_1}\left(A\frac{\partial \boldsymbol{w}}{\partial x_1}\right)\\ &\approx \frac{\left(A\frac{\partial \boldsymbol{w}}{\partial x_1}\right)|_{(x_1+\frac{\Delta x_1}{2},x_2,x_3)}-\left(A\frac{\partial \boldsymbol{w}}{\partial x_1}\right)|_{(x_1-\frac{\Delta x_1}{2},x_2,x_3)}}{\Delta x_1}\\ &= \frac{1}{\Delta x_1}\left[A(x_1+\frac{\Delta x_1}{2},x_2,x_3)\right.\\ &\quad\left.\times\left(\frac{\boldsymbol{w}(x_1+\Delta x_1,x_2,x_3)-\boldsymbol{w}(x_1,x_2,x_3)}{\Delta x_1}\right)\right]\\ &\quad-\frac{1}{\Delta x_1}\left[A(x_1-\frac{\Delta x_1}{2},x_2,x_3)\right.\\ &\quad\left.\times\left(\frac{\boldsymbol{w}(x_1,x_2,x_3)-\boldsymbol{w}(x_1-\Delta x_1,x_2,x_3)}{\Delta x_1}\right)\right]\end{aligned}$$

3. **VOXEL-CROSS DERIVATIVE:**

$$\begin{aligned}&\frac{\partial}{\partial x_2}\left(A\frac{\partial \boldsymbol{w}}{\partial x_1}\right)\\ &\approx \frac{\partial}{\partial x_2}\left(A(x_1,x_2,x_3)\right.\\ &\quad\left.\times\left(\frac{\boldsymbol{w}(x_1+\Delta x_1,x_2,x_3)-\boldsymbol{w}(x_1-\Delta x_1,x_2,x_3)}{2\Delta x_1}\right)\right)\\ &\approx \frac{1}{4\Delta x_1\Delta x_2}\times(A(x_1,x_2+\Delta x_2,x_3)\\ &\quad[\boldsymbol{w}(x_1+\Delta x_1,x_2+\Delta x_2,x_3)\\ &\quad-\boldsymbol{w}(x_1-\Delta x_1,x_2+\Delta x_2,x_3)]\\ &\quad-A(x_1,x_2-\Delta x_2,x_3)\,[\boldsymbol{w}(x_1+\Delta x_1,x_2-\Delta x_2,x_3)\\ &\quad-\boldsymbol{w}(x_1-\Delta x_1,x_2-\Delta x_2,x_3)])\end{aligned}$$

4. **VOXEL-INTERFACE:** $A(x_1 \pm \frac{\Delta x_1}{2},x_2,x_3) \approx \frac{1}{2}(A(x_1 \pm \Delta x_1,x_2,x_3)+A(x_1,x_2,x_3))$

3.3 Remarks on spatial derivatives

We have the following remarks:

Remark 2 To illustrate second-order accuracy, consider a Taylor series expansion for an arbitrary function $\boldsymbol{w}$

$$\begin{aligned}\boldsymbol{w}(x_1+\Delta x_1,x_2,x_3)=\boldsymbol{w}(x_1,x_2,x_3)+\frac{\partial \boldsymbol{w}}{\partial x_1}|_{(x_1,x_2,x_3)}\Delta x_1+\\ \frac{1}{2}\frac{\partial^2 \boldsymbol{w}}{\partial x_1^2}|_{(x_1,x_2,x_3)}(\Delta x_1)^2 x_1\\ +\frac{1}{6}\frac{\partial^3 \boldsymbol{w}}{\partial x_1^3}|_{(x_1,x_2,x_3)}(\Delta x_1)^3+\mathcal{O}((\Delta x_1)^4)\end{aligned} \tag{3.18}$$

and

$$\begin{aligned}\boldsymbol{w}(x_1-\Delta x_1,x_2,x_3)=\boldsymbol{w}(x_1,x_2,x_3)-\frac{\partial \boldsymbol{w}}{\partial x_1}|_{(x_1,x_2,x_3)}\Delta x_1+\\ \frac{1}{2}\frac{\partial^2 \boldsymbol{w}}{\partial x_1^2}|_{(x_1,x_2,x_3)}(\Delta x_1)^2 x_1\\ -\frac{1}{6}\frac{\partial^3 \boldsymbol{w}}{\partial x_1^3}|_{(x_1,x_2,x_3)}(\Delta x_1)^3+\mathcal{O}((\Delta x_1)^4).\end{aligned} \tag{3.19}$$

Subtracting the two expressions yields

$$\begin{aligned}&\frac{\partial \boldsymbol{w}}{\partial x_1}|_{(x_1,x_2,x_3)}\\ &\quad=\frac{\boldsymbol{w}(x_1+\Delta x_1,x_2,x_3)-\boldsymbol{w}(x_1-\Delta x_1,x_2,x_3)}{2\Delta x_1}\\ &\qquad+\mathcal{O}((\Delta x_1)^2).\end{aligned} \tag{3.20}$$

All other derivatives follow from this basic process, which is relatively standard in the all stencil-based discretizations.

Remark 3 At the length-scales of interest, when using voxels, it is questionable whether the ideas of a sharp material interface are justified. Accordingly, later, we simulated the system with and without Laplacian smoothing, whereby one smooths the material data by post-processing the original material data, voxel by voxel, to produce a smoother material representation, for example, for $\hat{A}$ (using the previous voxel approximations and nodal subscript notation):

$$\begin{aligned}\nabla^2 A=&\frac{1}{(\Delta x_i)^2}\left(A_{i+1,j,k}-2A_{i,j,k}+A_{i-1,j,k}\right)\\ &+\frac{1}{(\Delta x_j)^2}\left(A_{i,j+1,k}-2A_{i,j,k}+A_{i,j-1,k}\right)\\ &+\frac{1}{(\Delta x_k)^2}\left(A_{i,j,k+1}-2A_{i,j,k}+A_{i,j,k-1}\right)=0\end{aligned} \tag{3.21}$$

which yields a smoother value of $A_{i,j,k}$, denoted $\hat{A}_{i,j,k}$, given by

$$\begin{aligned}\nabla^2 A=0\Rightarrow \hat{A}_{i,j,k}=&\frac{1}{6}\left(A_{i+1,j,k}+A_{i-1,j,k}\right.\\ &\left.+A_{i,j+1,k}+A_{i,j-1,k}+A_{i,j,k+1}+A_{i,j,k-1}\right).\end{aligned} \tag{3.22}$$

The same process was applied to the other parameters, generically denoted, $A(\boldsymbol{x})$, by enforcing $\nabla_x^2 A=0$, as well as for any other material data. The results of the simulations, with and without data smoothing, were negligibly different for sufficiently fine voxel-meshes.

Remark 4 Solution fields for voxels that that fall outside of the domain are not computed, and are assigned no material values, which effectively enforces (correctly) a zero flux at a free surface, for example, $(\underbrace{\mathbb{E}}_{=\mathbf{0}}:\nabla\boldsymbol{u})\cdot\boldsymbol{n}=\mathbf{0}$.

3.4 Iterative (implicit) solution method

3.4.1 Recursion formula for second-order temporal discretization

Discretization of temporally second-order equations can be illustrated by considering

$$\ddot{\boldsymbol{U}}=\dot{\boldsymbol{V}}=\boldsymbol{L}(t). \tag{3.23}$$

where $\boldsymbol{U}$ is the (N-tuple) vector of all voxel displacement values and $\boldsymbol{V}$ is the vector of all voxel velocity values. Explicitly, $\boldsymbol{U}$

$$\boldsymbol{U}\overset{\text{def}}{=}\left\{\begin{array}{c}U_1\\U_2\\U_3\\...\\U_N\end{array}\right\} \tag{3.24}$$

and

$$\boldsymbol{V}\overset{\text{def}}{=}\left\{\begin{array}{c}V_1\\V_2\\V_3\\...\\V_N\end{array}\right\} \tag{3.25}$$

and

$$\boldsymbol{L}\overset{\text{def}}{=}\left\{\begin{array}{c}L_1\\L_2\\L_3\\...\\L_N\end{array}\right\} \tag{3.26}$$

where $\boldsymbol{L}$ represents all of the nodal values of $\boldsymbol{\Psi}$ found in Equation 3.4. Trapezoidal time-stepping discretization yields (detailed in Appendix 1)

$$\boldsymbol{V}(t+\Delta t) \approx \boldsymbol{V}(t) + \Delta t \boldsymbol{L}(t+\phi\Delta t) + \hat{\mathcal{O}}((\Delta t)^2) \tag{3.27}$$

where

$$\boldsymbol{L}(t+\phi\Delta t) \approx \phi\boldsymbol{L}(t+\Delta t) + (1-\phi)\boldsymbol{L}(t) + \hat{\mathcal{O}}((\Delta t)^2) \tag{3.28}$$

and

$$\boldsymbol{U}(t+\Delta t) \approx \boldsymbol{U}(t) + \Delta t \boldsymbol{V}(t+\phi\Delta t) + \hat{\mathcal{O}}((\Delta t)^2) \tag{3.29}$$

where

$$\boldsymbol{V}(t+\phi\Delta t) \approx (\phi\boldsymbol{V}(t+\Delta t) + (1-\phi)\boldsymbol{V}(t))\,\Delta t + \hat{\mathcal{O}}((\Delta t)^2), \tag{3.30}$$

which yields

$$\boldsymbol{U}(t+\Delta t) \approx \boldsymbol{U}(t) + \boldsymbol{V}(t)\Delta t + \phi(\Delta t)^2\,(\phi\boldsymbol{L}(t+\Delta t) + (1-\phi)\boldsymbol{L}(t)) + \hat{\mathcal{O}}((\Delta t)^2). \tag{3.31}$$

In iterative (recursion) form

$$\boldsymbol{U}^{L+1,K} \approx \underbrace{(\phi\Delta t)^2\boldsymbol{L}^{L+1,K-1}}_{\mathcal{G}(\boldsymbol{U}^{L+1,K-1})} + \underbrace{\boldsymbol{U}^L + \boldsymbol{V}^L\Delta t + (\Delta t)^2\phi(1-\phi)\boldsymbol{L}^L}_{\mathcal{R}}. \tag{3.32}$$

Implicit time-stepping methods, with time-step size adaptivity, built on approaches found in Zohdi [41–43], will be used throughout the upcoming analysis.

3.4.2 Error control and convergence

It is convenient to write Equation 3.32 in the following form

$$\mathcal{G}(\boldsymbol{U}^{L+1}) - \boldsymbol{U}^{L+1} + \mathcal{R} = \boldsymbol{0}, \tag{3.33}$$

where $\boldsymbol{U}^{L+1}$ is the vector value of all voxel values at time step $t+\Delta t$ and $\boldsymbol{U}^L$ is the vector value of all voxel values at time step t and where $\mathcal{R}$ is a remainder term that does not depend on the solution, i.e. $\mathcal{R} \neq \mathcal{R}(\boldsymbol{U}^{L+1})$. A straightforward iterative scheme can be written as

$$\boldsymbol{U}^{L+1,K} = \mathcal{G}(\boldsymbol{U}^{L+1,K-1}) + \mathcal{R}, \tag{3.34}$$

where $K = 1, 2, 3, \ldots$ is the index of iteration within time-step $L+1$. The convergence of such a scheme is dependent on the behavior of $\mathcal{G}$. Namely, a sufficient condition for convergence is that $\mathcal{G}$ is a contraction mapping for all $\boldsymbol{U}^{L+1,K}$, $K = 1, 2, 3\ldots$ In order to investigate this further, we define the iteration error as

$$\epsilon^{L+1,K} \stackrel{\text{def}}{=} \boldsymbol{U}^{L+1,K} - \boldsymbol{U}^{L+1}. \tag{3.35}$$

A necessary restriction for convergence is iterative self consistency, i.e. the "exact" (discretized) solution must be represented by the scheme

$$\mathcal{G}(\boldsymbol{U}^{L+1}) + \mathcal{R} = \boldsymbol{U}^{L+1}. \tag{3.36}$$

Enforcing this restriction, a sufficient condition for convergence is the existence of a contraction mapping

$$\begin{aligned}\epsilon^{L+1,K} &= ||\boldsymbol{U}^{L+1,K} - \boldsymbol{U}^{L+1}|| \\ &= ||\mathcal{G}(\boldsymbol{U}^{L+1,K-1}) - \mathcal{G}(\boldsymbol{U}^{L+1})|| \\ &\leq \eta^{L+1,K}||\boldsymbol{U}^{L+1,K-1} - \boldsymbol{U}^{L+1}||,\end{aligned} \tag{3.37}$$

where, if $0 \leq \eta^{L+1,K} < 1$ for each iteration K, then $\epsilon^{L+1,K} \rightarrow \boldsymbol{0}$ for any arbitrary starting value $\boldsymbol{U}^{L+1,K=0}$, as $K \rightarrow \infty$. This type of contraction condition is sufficient, but not necessary, for convergence. Recall, inserting the spatio-temporal approximations into Equation 3.23 leads to Equation 3.32, whose contraction constant is scaled by $\eta \propto (\Delta t)^p$. Therefore, if convergence is slow within a time-step, the time-step size, which is adjustable, can be reduced by an appropriate amount to increase the rate of convergence. Decreasing the time-step size improves the convergence, however, we want to simultaneously maximize the time-step sizes to decrease overall computing time, while still meeting an error tolerance on the numerical solution's accuracy. In order to achieve this goal, we follow an approach found in Zohdi [41–43] originally developed for continuum thermo-chemical multifield problems in which one first approximates

$$\eta^{L+1,K} \approx S(\Delta t)^p \tag{3.38}$$

(S is a constant) and secondly one assumes the error within an iteration to behave according to

$$(S(\Delta t)^p)^K \epsilon^{L+1,0} = \epsilon^{L+1,K}, \tag{3.39}$$

where $\epsilon^{L+1,0}$ is the initial norm of the iterative error and S is intrinsic to the system.[1] Our goal is to meet an error tolerance in exactly a preset number of iterations. To this end, one writes

$$(S(\Delta t_{\text{tol}})^p)^{K_d} \epsilon^{L+1,0} = C_{tol}, \tag{3.40}$$

[1] For the class of problems under consideration, due to the quadratic dependency on Δt, $p \approx 2$.

where C_{tol} is a (coupling) tolerance and where K_d is the number of desired iterations.[2] If the error tolerance is not met in the desired number of iterations, the contraction constant $\eta^{L+1,K}$ is too large. Accordingly, one can solve for a new smaller step size, under the assumption that S is constant,

$$\Delta t_{\text{tol}} = \Delta t \left(\frac{(\frac{C_{tol}}{\epsilon^{L+1,0}})^{\frac{1}{pK_d}}}{(\frac{\epsilon^{L+1,K}}{\epsilon^{L+1,0}})^{\frac{1}{pK}}} \right). \tag{3.41}$$

The assumption that S is constant is not critical, since the time-steps are to be recursively refined and unrefined throughout the simulation. Clearly, the expression in Equation 3.41 can also be used for time-step enlargement, if convergence is met in less than K_d iterations.[3] Specifically, the solution steps are, within a time-step (Figure 5):

- STEP 1: Start a global fixed iteration (set $i = 1, ..., N_n$ (voxel counter) and $K = 0$ (iteration counter))
- STEP 2: If $i > N_n$ then go to (4)
- STEP 3: If $i \leq N_n$ then:
  - (a)Compute the field $U_i^{L+1,K}$
  - (b)Go to STEP 2 for the next voxel ($i = i + 1$)
- STEP 4: Measure error (normalized)
  - (a) $\epsilon^{L+1,K} \stackrel{\text{def}}{=} \frac{\sum_{i=1}^{N_n} ||U_i^{L+1,K} - U_i^{L+1,K-1}||}{\sum_{i=1}^{N_n} ||U_i^{L+1,K}||}$
  - (b) $E_K \stackrel{\text{def}}{=} \frac{\epsilon^{L+1,K}}{TOL}$ where TOL is an error tolerance.
  - (c) $\Lambda_K \stackrel{\text{def}}{=} \left(\frac{(\frac{TOL}{\epsilon^{L+1,0}})^{\frac{1}{pK_d}}}{(\frac{\epsilon^{L+1,K}}{\epsilon^{L+1,0}})^{\frac{1}{pK}}} \right)$.
- STEP 5: If the tolerance is met: $E_K \leq 1$ and $K < K_d$ then
  - (a) Increment time: $t = t + \Delta t$
  - (b) Construct the next time-step: $(\Delta t)^{new} = \Lambda_K (\Delta t)^{old}$,
  - (c) Select the minimum size: $\Delta t = MIN((\Delta t)^{lim}, (\Delta t)^{new})$

  and go to STEP 1
- STEP 6: If the tolerance is not met: $E_K > 1$ and $K < K_d$ then
  - (a) Update the iteration counter: $K = K + 1$
  - (b) Reset the voxel counter: $i = 1$
  - (c) Go to STEP 2
- STEP 7: If the tolerance is not met ($E_K > 1$) and $K = K_d$ then
  - (a) Construct a new time-step: $(\Delta t)^{new} = \Lambda_K (\Delta t)^{old}$
  - (b) Restart at time t and go to STEP 1

A schematic of the overall solution scheme is shown in Figure 5.

3.5 A stress test example

As an example, consider a biaxial stress test shown in Figure 6, where each arm is pulled radially outwards. For illustration purposes, the deformations are magnified by a factor of 100. The normalized color coding indicates the level of which the L_2 norm of the deviatoric stress exceeds a threshold value. The cost of constructing an array for a temporal update using a voxel calculation is:

- $M \times V$, where V is the number of voxels and $M = \mathcal{O}(10)$ is associated with summing up the terms needed to construct the terms in $\nabla \cdot \boldsymbol{\sigma}$. Specifically, there are nine derivatives of the components of $\boldsymbol{\sigma}$, needed in $\rho \frac{\partial \boldsymbol{v}}{\partial t} = \nabla \cdot \boldsymbol{\sigma} + \boldsymbol{f}$ (Equation 3.17). To construct each component of $\boldsymbol{\sigma}$ one must perform four operations. Thus, M=36 operations in total.
- Thus, $K \times M \times V \approx 36KV$, is the total count per time-step, where K is the number of iterations in a time step.

The cost of constructing an array for temporal update using an FEM calculation is associated with (1) meshing structure (2) numerically integrating the weak form (3) generating a stiffness matrix and (4) solving a system of equations. Specifically (Zohdi [44]):

- Construction of the stiffness matrix: $P \times E$, where E is the number of elements and $P = \mathcal{O}(3000)$ stems from mapping and integrating the terms needed to construct the stiffness matrix associated with the weak form of $\nabla \cdot \boldsymbol{\sigma}$ (of order $\mathcal{O}(10)$), of which there are 300 entries in a (symmetric) linear elasticity element stiffness matrix (linear hexahedra), yielding $P = 300 \times 10 = 3000$.
- Iterative solution: $I \times Q$, where I is the number of iterations associated with, for example, a Conjugate-Gradient solver and Q is the cost of a matrix-vector multiplication. Q is on the order of N^q, where N^q is the number of nodes in the system and $1 < q \leq 2$. For example, for linearized elasticity, using an element-by-element multiplier (not counting preconditioning), for linearized elasticity and using linear hexahedra, $Q = 300N$. Thus, $I \times Q \approx 300IN$.
- Thus, a comparison of the total operation counts between the Voxel method and FEM is roughly

$$\frac{Voxel}{FEM} \approx \frac{36KV}{3000E + 300IN}. \tag{3.42}$$

Equation 3.42 clearly shows the favorable ratio of operation counts of the voxel approach relative to FEM. In more complex (nonlinear) problems, where the stiffness matrix would have to be reformed after each iteration, the term $3000E$ would need to be multiplied by I.

3.5.1 Remarks on iterative schemes

We have the following remarks:

Remark 5 Applying this scheme to the balance of linear momentum continuum formulation, under infinitesimal deformations, $\nabla_x \cdot \boldsymbol{\sigma} + \boldsymbol{f} = \rho \frac{\partial^2 \boldsymbol{u}}{\partial t^2}$ we use $\boldsymbol{\Psi}(\boldsymbol{u}(t)) = \frac{\nabla_x \cdot \boldsymbol{\sigma} + \boldsymbol{f}}{\rho}$, and must apply the (iterative) process introduced

[2] Typically, K_d is chosen to be between five to ten iterations, although this is problem and analyst dependent.

[3] At the implementation level, since the exact solution is unknown, the following relative error term is used, $\epsilon^{L+1,K} \stackrel{\text{def}}{=} \boldsymbol{U}^{L+1,K} - \boldsymbol{U}^{L+1,K-1}$.

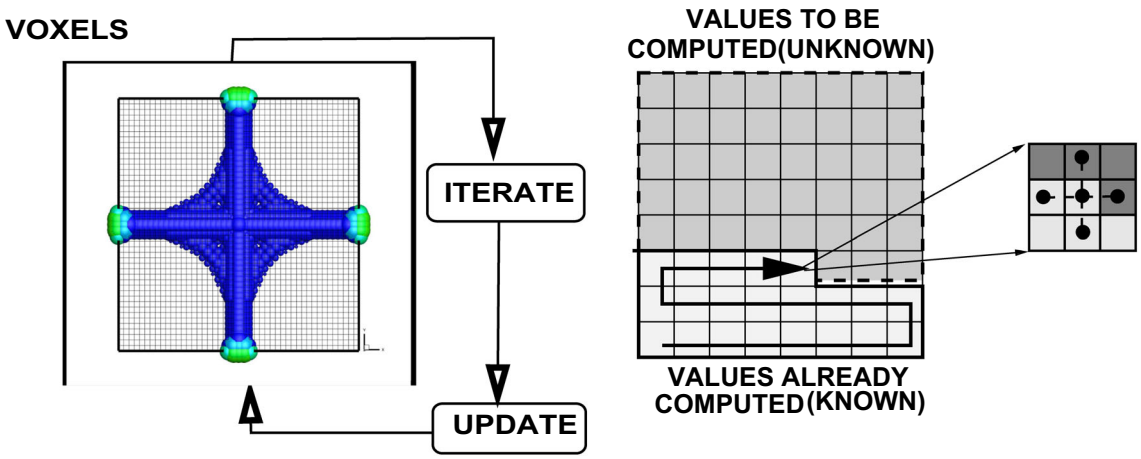

Fig. 5 The overall iterative (left) solution and the matrix-free approach using a moving front through the voxels (right). During the iterative solution process, the most current value of a voxel is used in any calculation, for example a construction of the Laplacian, or any other term in the governing differential equations

earlier to all voxels in the system. Time-step size adaptivity is critical, since the system's dynamics can dramatically change over the course of time, possibly requiring quite different time-step sizes to control the iterative error. However, to maintain the accuracy of the time-stepping scheme, one must respect an upper bound dictated by the discretization error, i.e., $\Delta t \leq \Delta t^{lim}$. Note that in step (5), Λ_K may enlarge the time-step if the error is lower than the preset tolerance. At a given time, once the process is complete, the time is incremented forward and the process is repeated. The overall goal is to deliver solutions where the iterative error is controlled and the temporal discretization accuracy dictates the upper limit on the time-step size (Δt^{lim}). Clearly, there are various combinations of solution methods that one can choose from. For example, one may choose implicit ($0 < \phi \leq 1$) or explicit time-stepping ($\phi = 0$), and, in the case of implicit time-stepping, iterative or direct solvers. Furthermore, one could employ internal iterations for each field equation, then update, more sophisticated metrics for certain components of the error, etc. For example, we utilized an error measure that used the fields at the voxels of the Finite Difference grid, but other metrics are certainly possible. For details see Zohdi [41–43].

Remark 6 Because the internal system solvers within the staggering scheme are also iterative and use the previously converged solution as their starting value to solve the system of equations, a field that is relatively insensitive at given stage of the simulation will converge in very few internal iterations (perhaps even one). Staggering schemes are widely used in the computational mechanics literature, dating back, at least, to Zienkiewicz [45] and Zienkiewicz et al. [46]. For in depth overviews, see the works of Lewis and Schrefler (Lewis et al. [47] and Lewis and Schrefler [48]) and a series of works by Schrefler and collaborators: Schrefler [49], Turska and Schrefler [50], Bianco et al [72] and Wang and Schrefler [52].

Remark 7 During the iterative solution process, the most current value of a voxel is used in any calculation, for example a construction of the Laplacian, or any other term in the governing differential equations.

4 Tailored structural material mixtures for drones

In drone applications, there is growing industrial interest in using new microscale particle-enhanced composite materials (Figure 7). Typically, and easy to form thermoplastic (such as lexan, ABS, nylon, etc.) with added metallic or ceramic particles to stiffen (functionalize) the overall material. During the design development of such structures, such as the chassis, it is advantageous to ascertain the stress load-sharing between the added particles and the binding matrix, in order to make estimates of the structure's useful life and the material performance. Accordingly, a relatively straightforward approach is correlate the phase-averaged microstructural stress levels carried by the particles and matrix to the macrostructural loading. This is done by taking macroscale stresses, determined by using a structural-scale model with homogenized properties, with a microscale stresses determined from microstress concentration functions. This provides analysts with an easy to use design framework that clearly identifies the stress contributions from the microscale and the macroscale, in order to reduce product development time and costs.

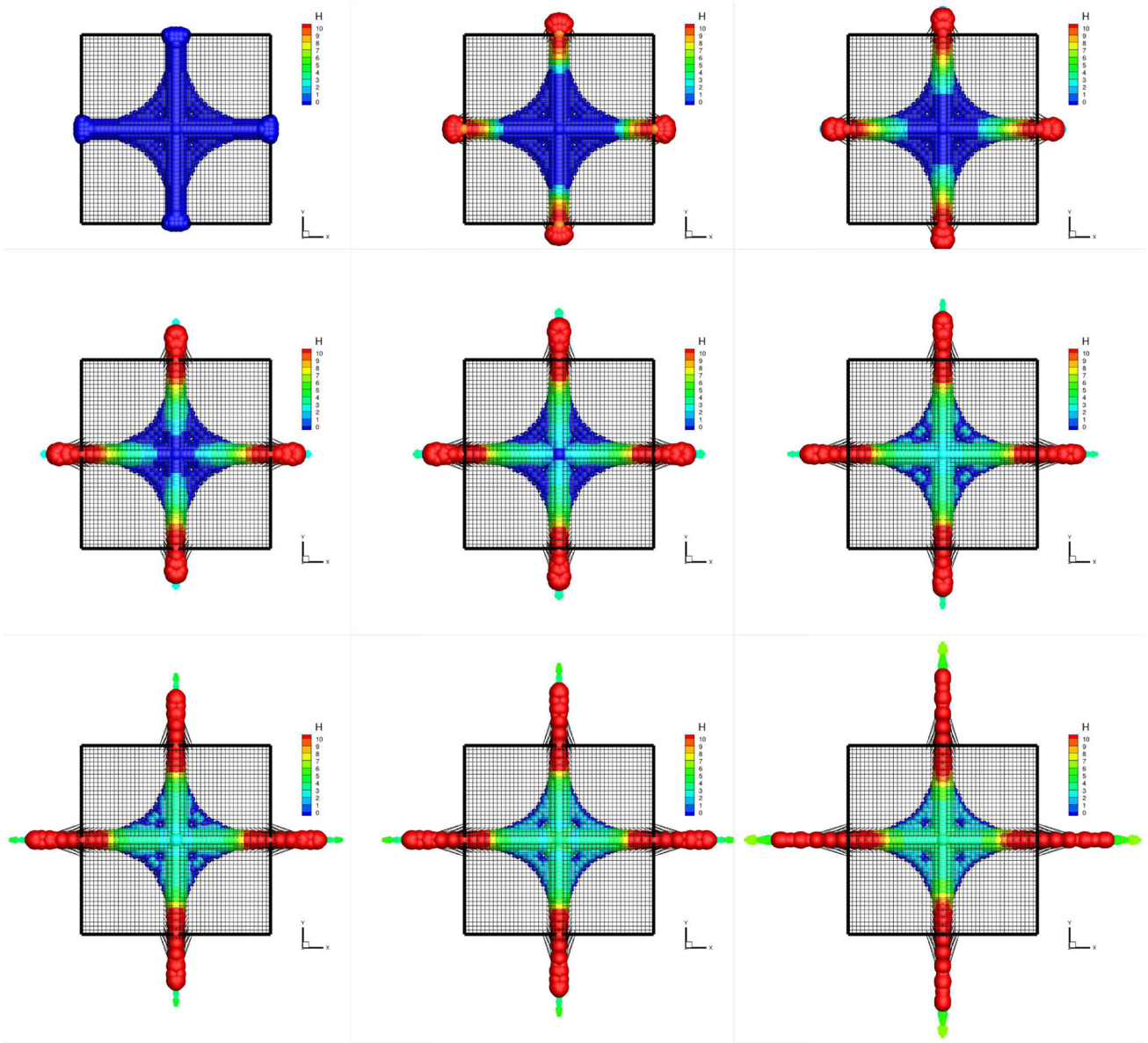

Fig. 6 An example: a dynamic biaxial pull test where each arm is pulled radially outwards. For illustration purposes, the deformations are magnified by a factor of 100. The normalized color coding indicates the level of which the L_2 norm of the deviatoric stress exceeds a threshold value

Due to a wide-range of technological advancements, in particular in advanced manufacturing, analysts are now afforded with many particle-matrix choices for structural materials. However, due to the nature of such applications, experiments to determine the appropriate combinations of particle and matrix materials are time-consuming and expensive, and it is advantageous to characterize such materials analytically and computationally, in order to reduce product development time and costs.

In order to characterize the effective macroscale (structural) material response of such materials, a relation between averages,

$$\langle \boldsymbol{\sigma} \rangle_\Omega = \mathbb{E}^* : \langle \boldsymbol{\epsilon} \rangle_\Omega, \tag{4.1}$$

is sought, where

$$\langle \cdot \rangle_\Omega \stackrel{\text{def}}{=} \frac{1}{|\Omega|} \int_\Omega \cdot \, d\Omega \,, \tag{4.2}$$

and where, throughout the structure, the mechanical properties of microheterogeneous materials are characterized by a spatially variable elasticity tensor $\mathbb{E} = \mathbb{E}(\boldsymbol{x})$ and $\boldsymbol{\sigma}$ and $\boldsymbol{\epsilon}$ are the stress and strain tensor fields within a Representative Volume Element (RVE) of volume $|\Omega|$. The quantity $\mathbb{E}^*$ is known as the effective property. It is the elasticity tensor used

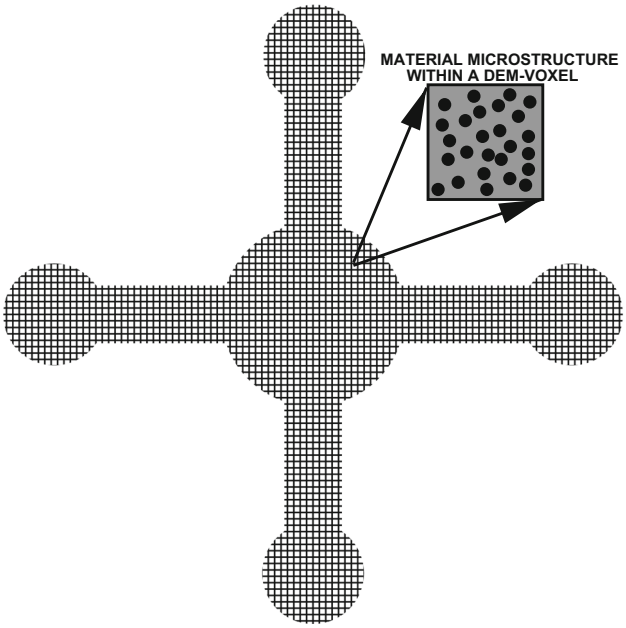

Fig. 7 An example: a microstructure comprised of a matrix binder and functionalizing particulate additives

in usual structural (macroscale) analyses. The internal fields, which are to be volumetrically averaged, can be computed by solving a series of boundary value problems with test loadings over an RVE using the Finite Element Method. However, this is extremely computationally intensive (Zohdi and Wriggers [53]) and oftentimes, faster approximate methods are sought. *Rapid approximations are important as a design tool.* Accordingly, we correlate the phase-averaged microstructural stress levels carried by the particles and matrix to the macrostructural loading. The objective is to provide analysts with an easy to use design framework that clearly identifies the stress load-sharing contributions at the microscale, in order to reduce product development time and costs.

Remark 8 We will concentrate on isotropic materials in this work. In the case of isotropic overall responses, we may write

$$\langle \frac{tr\boldsymbol{\sigma}}{3} \rangle_{\Omega} = 3\kappa^* \langle \frac{tr\boldsymbol{\epsilon}}{3} \rangle_{\Omega} \tag{4.3}$$

and

$$\langle \boldsymbol{\sigma}' \rangle_{\Omega} = 2\mu^* \langle \boldsymbol{\epsilon}' \rangle_{\Omega}, \tag{4.4}$$

where κ^* and μ^* are the effective bulk and shear moduli, $\frac{tr\boldsymbol{\sigma}}{3}$ is the dilatational stress, $\frac{tr\boldsymbol{\epsilon}}{3}$ is the dilatational strain, $\boldsymbol{\sigma}'$ is the deviatoric stress and $\boldsymbol{\epsilon}'$ is the deviatoric strain.

Remark 9 For an authoritative review of the general theory of random heterogeneous media see, for example, see Torquato [54] for general interdisciplinary discussions, Jikov et. al. [55] for more mathematical aspects, Hashin [56], Mura [57] or Markov [58] for solid-mechanics inclined accounts of the subject, for analyses of defect-laden, porous and cracked media, see Kachanov [59], Kachanov, Tsukrov and Shafiro [60], Kachanov and Sevostianov [61], Sevostianov, Gorbatikh and Kachanov [62], Sevostianov and Kachanov [63], and for computational aspects see Ghosh [64], Ghosh and Dimiduk [65], Zohdi [66], Zohdi and Wriggers [53].

Remark 10 There exist several modern techniques manufacturing composite materials for deployment in complex structural shapes, such as sacrificial patterning (Singh and Singh [68]), reinforcement with carbon nanotubes (Isaza et al [69]), multiphase extrusion (Khalifa et al [70]), microcutting arrays (Pacella et al [71]), interlaminar glass reinforcement (Bian et al [72]), electric field assistance (Decker and Gan [73], Zohdi [74]) and selective laser sintering (Gu et al [75]), where integration of the upcoming approach developed in this work could be useful.

4.0.2 Effective property estimates

The literature on methods to estimate the overall macroscopic properties of heterogeneous materials dates back at least to the 1800's by the pioneering works of Maxwell [76, 77] and Lord Rayleigh [78], with an extremely important contribution being the Hashin-Shtrikman bounds during the 1960's (Hashin and Shtrikman [79, 80], Hashin [56]). The Hashin-Shtrikman bounds are the tightest possible bounds on isotropic effective responses, generated from isotropic microstructures, where the volumetric data and phase contrasts of the constituents are the only data known. They are as follows, for the bulk modulus:

$$\begin{aligned} \kappa^{*,-} \stackrel{\text{def}}{=} \kappa_1 + \frac{v_2}{\frac{1}{\kappa_2-\kappa_1}+\frac{3(1-v_2)}{3\kappa_1+4\mu_1}} \leq \kappa^* \leq \kappa_2 \\ + \frac{1-v_2}{\frac{1}{\kappa_1-\kappa_2}+\frac{3v_2}{3\kappa_2+4\mu_2}} \stackrel{\text{def}}{=} \kappa^{*,+}, \end{aligned} \tag{4.5}$$

and for the shear modulus

$$\begin{aligned} \mu^{*,-} \stackrel{\text{def}}{=} \mu_1 + \frac{v_2}{\frac{1}{\mu_2-\mu_1}+\frac{6(1-v_2)(\kappa_1+2\mu_1)}{5\mu_1(3\kappa_1+4\mu_1)}} \leq \mu^* \leq \mu_2 \\ + \frac{(1-v_2)}{\frac{1}{\mu_1-\mu_2}+\frac{6v_2(\kappa_2+2\mu_2)}{5\mu_2(3\kappa_2+4\mu_2)}} \stackrel{\text{def}}{=} \mu^{*,+}, \end{aligned} \tag{4.6}$$

where κ_2 and κ_1 are the bulk moduli and μ_2 and μ_1 are the shear moduli of the respective phases ($\kappa_2 \geq \kappa_1$ and $\mu_2 \geq \mu_1$), and where v_2 is the second phase volume fraction. Phase 2 is the stiffer of the two constituents (which usually corresponds to the particles)[4]. One can form estimates for the effective properties by forming a convex combination of them, such as

[4] Note that no geometric or statistical information is required for the bounds.

$$\kappa^* \approx \phi\kappa^{*,+} + (1-\phi)\kappa^{*,-} \tag{4.7}$$

and

$$\mu^* \approx \phi\mu^{*,+} + (1-\phi)\mu^{*,-}, \tag{4.8}$$

where $0 \leq \phi \leq 1$ is a parameter such that:

- If $\phi = 0$ we have the lower bound,
- If $\phi = 1$ we have the upper bound and
- If $\phi = 1/2$ we have the average of the bounds.

Although the usual approach is to select $\phi = 0.5$, a more detailed discussion on the sharper selection of ϕ is provided in Appendices 1 and 2.

Remark 11 If needed, one can post-process the effective bulk and shear modulus to obtain the effective Poisson ratio $\nu^* = \frac{3\kappa^* - 2\mu^*}{2(3\kappa^* + \mu^*)}$ and the effective Young's modulus $E^* = 2\mu^*(1+\nu^*) = 3\kappa^*(1-2\nu^*)$.

4.0.3 Load-sharing between the particles and the matrix

The determination of the load-sharing between phases at the microstructral scale can be obtained by post-processing the overall effective mechanical properties of the microheterogeneous material. Specifically, one can decompose average of the stress over Ω into averages over each of the phases in the following manner:

$$\langle\boldsymbol{\sigma}\rangle_\Omega = \frac{1}{|\Omega|}\left(\int_{\Omega_1} \boldsymbol{\sigma}\, d\Omega + \int_{\Omega_2} \boldsymbol{\sigma}\, d\Omega\right) = v_1\langle\boldsymbol{\sigma}\rangle_{\Omega_1} + v_2\langle\boldsymbol{\sigma}\rangle_{\Omega_2}, \tag{4.9}$$

where Ω_1 is the domain of phase 1 (usually the matrix) and Ω_2 is the domain of phase 2 (usually the particles). If we make use of this decomposition for the stresses, we have

$$\begin{aligned}\langle\boldsymbol{\sigma}\rangle_\Omega &= v_1\langle\boldsymbol{\sigma}\rangle_{\Omega_1} + v_2\langle\boldsymbol{\sigma}\rangle_{\Omega_2}\\ &= v_1\mathbb{E}_1 : \langle\boldsymbol{\epsilon}\rangle_{\Omega_1} + v_2\mathbb{E}_2 : \langle\boldsymbol{\epsilon}\rangle_{\Omega_2}\\ &= \mathbb{E}_1 : (\langle\boldsymbol{\epsilon}\rangle_\Omega - v_2\langle\boldsymbol{\epsilon}\rangle_{\Omega_2}) + v_2\mathbb{E}_2 : \langle\boldsymbol{\epsilon}\rangle_{\Omega_2}\\ &= \left(\mathbb{E}_1 + v_2(\mathbb{E}_2 - \mathbb{E}_1) : \boldsymbol{C}^{\epsilon,2}\right) : \langle\boldsymbol{\epsilon}\rangle_\Omega,\end{aligned} \tag{4.10}$$

where

$$\boldsymbol{C}^{\epsilon,2} \overset{\text{def}}{=} \left(\frac{1}{v_2}(\mathbb{E}_2 - \mathbb{E}_1)^{-1} : (\mathbb{E}^* - \mathbb{E}_1)\right) \tag{4.11}$$

is called the strain concentration function, with $\boldsymbol{C}^{\epsilon,2} : \langle\boldsymbol{\epsilon}\rangle_\Omega = \langle\boldsymbol{\epsilon}\rangle_{\Omega_2}$. The strain concentration tensor $\boldsymbol{C}^{\epsilon,2}$ relates the average strain over the particle phase (2) to the average strain over all phases. Similarly, for the variation in the stress we have $\boldsymbol{C}^{\epsilon,2} : \mathbb{E}^{*-1} : \langle\boldsymbol{\sigma}\rangle_\Omega = \mathbb{E}_2^{-1} : \langle\boldsymbol{\sigma}\rangle_{\Omega_2}$, which reduces to $\mathbb{E}_2 : \boldsymbol{C}^{\epsilon,2} : \mathbb{E}^{*-1} : \langle\boldsymbol{\sigma}\rangle_\Omega \overset{\text{def}}{=} \boldsymbol{C}^{\sigma,2} : \langle\boldsymbol{\sigma}\rangle_\Omega = \langle\boldsymbol{\sigma}\rangle_{\Omega_2}$. $\boldsymbol{C}^{\sigma,2}$ is known as the stress concentration tensor and it relates the average stress in the particle phase to that in the whole RVE. Note that once either $\boldsymbol{C}^{\epsilon,2}$ or $\mathbb{E}^*$ are known, the other can be determined. In the case of isotropy we may write (directly from Equation 4.11)

$$C_\kappa^{\sigma,2} \overset{\text{def}}{=} \frac{1}{v_2}\frac{\kappa_2}{\kappa^*}\frac{\kappa^* - \kappa_1}{\kappa_2 - \kappa_1} \qquad \text{and} \qquad C_\mu^{\sigma,2} \overset{\text{def}}{=} \frac{1}{v_2}\frac{\mu_2}{\mu^*}\frac{\mu^* - \mu_1}{\mu_2 - \mu_1} \tag{4.12}$$

where $C_\kappa^{\sigma,2}\langle\frac{tr\boldsymbol{\sigma}}{3}\rangle_\Omega = \langle\frac{tr\boldsymbol{\sigma}}{3}\rangle_{\Omega_2}$ and where $C_\mu^{\sigma,2}\langle\boldsymbol{\sigma}'\rangle_\Omega = \langle\boldsymbol{\sigma}'\rangle_{\Omega_2}$. Clearly, the microstress fields are minimally distorted when $C_\kappa^{\sigma,2} = C_\mu^{\sigma,2} = 1$ (for example, there are no stress concentrations in a homogeneous material). For the matrix, we write

$$\begin{aligned}\langle\boldsymbol{\sigma}\rangle_{\Omega_1} &= \frac{\langle\boldsymbol{\sigma}\rangle_\Omega - v_2\langle\boldsymbol{\sigma}\rangle_{\Omega_2}}{v_1} = \frac{\langle\boldsymbol{\sigma}\rangle_\Omega - v_2\boldsymbol{C}^{\sigma,2}:\langle\boldsymbol{\sigma}\rangle_\Omega}{v_1}\\ &= \frac{(\mathbf{1} - v_2\boldsymbol{C}^{\sigma,2}):\langle\boldsymbol{\sigma}\rangle_\Omega}{v_1} \overset{\text{def}}{=} \boldsymbol{C}^{\sigma,1} : \langle\boldsymbol{\sigma}\rangle_\Omega.\end{aligned} \tag{4.13}$$

In the case of isotropy,

$$C_\kappa^{\sigma,1} \overset{\text{def}}{=} \frac{1}{v_1}(1 - v_2C_\kappa^{\sigma,2}) \qquad \text{and} \qquad C_\mu^{\sigma,1} \overset{\text{def}}{=} \frac{1}{v_1}(1 - v_2C_\mu^{\sigma,2}). \tag{4.14}$$

The portion of the total stress carried by each phase can be then determined by multiplying the concentration factors by the corresponding volume fractions (using Equation 4.9)

$$\begin{aligned}\langle\boldsymbol{\sigma}\rangle_\Omega = v_1\langle\boldsymbol{\sigma}\rangle_{\Omega_1} + v_2\langle\boldsymbol{\sigma}\rangle_{\Omega_2} = v_1\boldsymbol{C}^{\sigma,1} : \langle\boldsymbol{\sigma}\rangle_\Omega + v_2\boldsymbol{C}^{\sigma,2} :\\ \langle\boldsymbol{\sigma}\rangle_\Omega = (v_1\boldsymbol{C}^{\sigma,1} + v_2\boldsymbol{C}^{\sigma,2}) : \langle\boldsymbol{\sigma}\rangle_\Omega.\end{aligned} \tag{4.15}$$

In the case of isotropy

$$\langle\boldsymbol{\sigma}\rangle_\Omega = (v_1C^{\sigma,1} + v_2C^{\sigma,2})\langle\boldsymbol{\sigma}\rangle_\Omega \Rightarrow v_1C^{\sigma,1} + v_2C^{\sigma,2} = 1. \tag{4.16}$$

4.0.4 Example

Consider the following parameters, for a hypothetical drone material mixture of ABS (plastic, phase 1) and steel particles (phase 2):

- The volume fraction of particles: $v_2 = 0.5$,
- The bound averaging parameter: $\phi = 0.5$,
- The shear moduli, ABS, $\mu_1 = 2$ GPa and steel, $\mu_2 = 79$ GPa and
- The bulk moduli, ABS, $\kappa_1 = 3.8$ GPa and steel $\kappa_2 = 160$ GPa.
- The density, ABS $\rho_1 = 1005\,\text{kg}/m^3$ and steel, $\rho_2 = 7850\,\text{kg}/m^3$,

The following quantities result from this parameter selection:

- The predicted overall shear modulus: $\mu^* = 17.252$ Gpa
- The predicted overall bulk modulus: $\kappa^* = 29.547$ Gpa
- The predicted overall density: $\rho^* = 4427\ kg/m^3$
- The deviatoric stress concentration in phase 1: $C_\mu^{\sigma,1}$= 0.1859
- The deviatoric stress concentration in phase 2: $C_\mu^{\sigma,2}$= 1.8140
- The dilatational stress concentration in phase 1: $C_\kappa^{\sigma,1}$= 0.2148
- The dilatational stress concentration in phase 2: $C_\kappa^{\sigma,2}$= 1.7851
- The fraction of the deviatoric stress carried by phase 1: $v_1 C_\mu^{\sigma,1}$=0.0929
- The fraction of the deviatoric stress carried by phase 2: $v_2 C_\mu^{\sigma,2}$=0.9070
- The fraction of the dilatational stress carried by phase 1:$v_1 C_\kappa^{\sigma,1}$=0.1074
- The fraction of the dilatational stress carried by phase 2: $v_2 C_\kappa^{\sigma,1}$=0.8925.

The previous analysis is motivated by the increasing use of tailored particle-enhanced materials in drone chassis applications. This analysis correlated the phase-averaged microstructural stresses experienced by material to the macrostructural loading. This provides a framework for an easy to use design tool that clearly identifies the stress load share contributions, which aids in reducing product development time and cost. Of course, one can resort to a multiscale FEM analysis (Zohdi and Wriggers [53]), but the computational costs are prohibitive. More on these estimates can be found in Appendices 1 and 2.

Remark 12 The next-stage in the overall work flow is to deploy the drone "into the field" for flight tests. In this vein, we follow a process found in Zohdi [28], which utilizes the Discrete Element Method (DEM) for detailed rigid-body drone dynamics.

5 Voxel-based discrete elements for dynamic drone flight evaluation

To assess the dynamic performance of a voxel-generated quadcopter, we connect the previous methods to a rapid structural dynamics simulator, following Zohdi [28], based on the Discrete Element Method (DEM). In Figure 8, we show the conversion from raw voxels for stress analysis to discrete elements for dynamic flight analysis. For $(p_1, p_2, p_3) = (0.5, 0.5, 0.5)$, a DEM generated drone frame is shown with a zoom on the *locations* of the DEM particles. The particles are bound by the mathematical dynamics-constraints to move collectively as a rigid body (group translation and rotation), unless a particle is dislodged by excessive force. If dislodged, the fragment moves according to its own dynamics. As an example of how this is used in conjunction with rapid structural design, we illustrate the deployment and dynamic performance evaluation of tactical quadcopter drones under attack, specifically by being subjected to series of launched explosions. In that vein, the DEM is used to ascertain the response of a drone to a hostile environment, in order to ascertain its deployed performance in the field. The method also allows one to potentially describe damage to the quadcopter drone, its loss of functionality (thrust), etc. Furthermore, the use of the DEM can also allow for fragmentation of the quadcopter and can also ascertain the resulting debris field (Zohdi [28]). The procedure to convert voxels into discrete elements is straightforward (Figure 8):

- Assign a spherical mass and moment of inertia to the center of each voxel
- The mass is proportional to the mass in the voxel
- The mass moment of inertia is proportional to mass and size of the DEM sphere
- The masses move according to the dynamics of a rigid body, which is subject to Newton's laws

5.1 Group dynamics of a rigidly bound collection of drone particles/elements

in order to make the analysis general, we consider rigid clusters of DEM particles. Later we will tailor the cluster to specific drone designs. We consider the DEM cluster to be already formed, with particles rigidly bonded together. Later, we will allow particles to become dislodged from the cluster. Consider a collection of rigidly-bonded particles, $i = 1, 2, ..., N_c$, in a cluster. The individual particle dynamics are described by (which leads to a coupled system of ODE's)

$$m_i \ddot{\boldsymbol{r}}_i = m_i \dot{\boldsymbol{v}}_i = \underbrace{\boldsymbol{\Psi}_i^{tot}}_{\text{total forces}} = \underbrace{\boldsymbol{\Psi}_i^{int}}_{\text{internal}} + \underbrace{\boldsymbol{\Psi}_i^{ext}}_{\text{external}}, \tag{5.1}$$

where m_i is the mass of the ith particle, $\boldsymbol{r}_i$ is the position vector, $\boldsymbol{v}_i$ is the particle velocity, $\boldsymbol{\Psi}_i^{ext}$ is an external force field and $\boldsymbol{\Psi}_i^{int}$ is the sum of the internal (equal in magnitude and opposite in direction) forces acting on the ith particle, due to other particles in the system ("internal" particle-to-particle bonding forces, contact forces etc. When we consider a collection of particles that are bound together as a rigid body, because internal forces between particles within in the system are opposite in direction and equal in magnitude, the specific character of the internal particle-to-particle bonding forces is not relevant to the overall system dynamics,

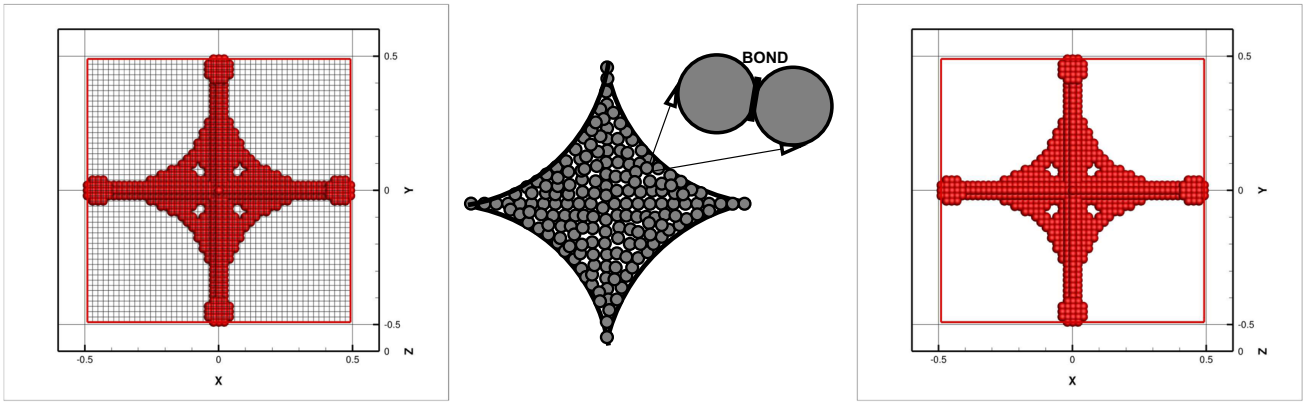

Fig. 8 Conversion from raw voxels for stress analysis to discrete elements for dynamic flight analysis. The particles are bound by the mathematical dynamics-constraints to move collectively as a rigid body (group translation and rotation)

$$\sum_{i=1}^{N_c}\left(\boldsymbol{\Psi}_i^{ext}+\boldsymbol{\Psi}_i^{int}\right)=\sum_{i=1}^{N_c}\boldsymbol{\Psi}_i^{ext}+\underbrace{\sum_{i=1}^{N_c}\boldsymbol{\Psi}_i^{int}}_{=\mathbf{0}}$$
$$=\sum_{i=1}^{N_c}\boldsymbol{\Psi}_i^{ext}\overset{\text{def}}{=}\boldsymbol{\Psi}^{EXT}, \tag{5.2}$$

where $\boldsymbol{\Psi}^{EXT}$ is the overall external force acting on the cluster and N_c are the number of particles in the DEM cluster. The position vector of the center of mass of the system is given by

$$\boldsymbol{r}_{cm}\overset{\text{def}}{=}\frac{\sum_{i=1}^{N_c}m_i\boldsymbol{r}_i}{\sum_{i=1}^{N_c}m_i}=\frac{1}{\mathcal{M}}\sum_{i=1}^{N_c}m_i\boldsymbol{r}_i, \tag{5.3}$$

where $\mathcal{M}$ is the total system mass. A decomposition of the position vector for particle i, of the form $\boldsymbol{r}_i=\boldsymbol{r}_{cm}+\boldsymbol{r}_{cm\to i}$, allows the linear momentum of the system of particles ($\boldsymbol{G}$) to be written as

$$\sum_{i=1}^{N_c}\underbrace{m_i\dot{\boldsymbol{r}}_i}_{\boldsymbol{G}_i}=\sum_{i=1}^{N_c}m_i(\dot{\boldsymbol{r}}_{cm}+\dot{\boldsymbol{r}}_{cm\to i})=\sum_{i=1}^{N_c}m_i\dot{\boldsymbol{r}}_{cm}$$
$$=\dot{\boldsymbol{r}}_{cm}\sum_{i=1}^{N_c}m_i=\mathcal{M}\dot{\boldsymbol{r}}_{cm}\overset{\text{def}}{=}\boldsymbol{G}_{cm}, \tag{5.4}$$

since $\sum_{i=1}^{N_c}m_i\dot{\boldsymbol{r}}_{cm\to i}=\mathbf{0}$. Furthermore, $\dot{\boldsymbol{G}}_{cm}=\mathcal{M}\ddot{\boldsymbol{r}}_{cm}$, thus

$$\dot{\boldsymbol{G}}_{cm}=\mathcal{M}\ddot{\boldsymbol{r}}_{cm}=\sum_{i=1}^{N_c}\boldsymbol{\psi}_i^{ext}\overset{\text{def}}{=}\boldsymbol{\Psi}^{EXT}. \tag{5.5}$$

The angular momentum relative to the center of mass can be written as (utilizing $\dot{\boldsymbol{r}}_i=\boldsymbol{v}_i=\boldsymbol{v}_{cm}+\boldsymbol{v}_{cm\to i}$)

$$\sum_{i=1}^{N_c}\boldsymbol{H}_{cm\to i}=\sum_{i=1}^{N_c}(\boldsymbol{r}_{cm\to i}\times m_i\boldsymbol{v}_{cm\to i})$$
$$=\sum_{i=1}^{N_c}(\boldsymbol{r}_{cm\to i}\times m_i(\boldsymbol{v}_i-\boldsymbol{v}_{cm})) \tag{5.6}$$
$$=\sum_{i=1}^{N_c}(m_i\boldsymbol{r}_{cm\to i}\times\boldsymbol{v}_i)-\left(\underbrace{\sum_{i=1}^{N_c}m_i\boldsymbol{r}_{cm\to i}}_{=0}\right)\times\boldsymbol{v}_{cm}=\boldsymbol{H}_{cm}, \tag{5.7}$$

for a rigid body. Since $\boldsymbol{v}_{cm\to i}=\boldsymbol{\omega}\times\boldsymbol{r}_{cm\to i}$

$$\boldsymbol{H}_{cm}=\sum_{i=1}^{N_c}\boldsymbol{H}_{cm\to i}=\sum_{i=1}^{N_c}m_i(\boldsymbol{r}_{cm\to i}\times\boldsymbol{v}_{cm\to i})$$
$$=\sum_{i=1}^{N_c}m_i(\boldsymbol{r}_{cm\to i}\times(\boldsymbol{\omega}\times\boldsymbol{r}_{cm\to i})). \tag{5.8}$$

Decomposing the relative position vector into its components

$$\boldsymbol{r}_{cm\to i}=\boldsymbol{r}_i-\boldsymbol{r}_{cm}=\hat{x}_{i1}\boldsymbol{e}_1+\hat{x}_{i2}\boldsymbol{e}_2+\hat{x}_{i3}\boldsymbol{e}_3, \tag{5.9}$$

where $\hat{x}_{i1}$, $\hat{x}_{i2}$ and $\hat{x}_{i3}$ are the coordinates of the mass points measured *relative to the center of mass*, and expanding the angular momentum expression, yields

$$H_1=\omega_1\sum_{i=1}^{N_c}(\hat{x}_{i2}^2+\hat{x}_{i3}^2)\,m_i-\omega_2\sum_{i=1}^{N_c}\hat{x}_{i1}\hat{x}_{i2}\,m_i-\omega_3\sum_{i=1}^{N_c}\hat{x}_{i1}\hat{x}_{i3}\,m_i \tag{5.10}$$

and

$$H_2 = -\omega_1 \sum_{i=1}^{N_c} \hat{x}_{i1}\hat{x}_{i2}\, m_i + \omega_2 \sum_{i=1}^{N_c} (\hat{x}_{i1}^2 + \hat{x}_{i3}^2)\, m_i - \omega_3 \sum_{i=1}^{N_c} \hat{x}_{i2}\hat{x}_{i3}\, m_i \tag{5.11}$$

and

$$H_3 = -\omega_1 \sum_{i=1}^{N_c} \hat{x}_{i1}\hat{x}_{i3}\, m_i - \omega_2 \sum_{i=1}^{N_c} \hat{x}_{i2}\hat{x}_{i3}\, m_i + \omega_3 \sum_{i=1}^{N_c} (\hat{x}_{i1}^2 + \hat{x}_{i2}^2)\, m_i, \tag{5.12}$$

which can be concisely written as

$$\boldsymbol{H}_{cm} = \overline{\mathcal{I}} \cdot \boldsymbol{\omega}, \tag{5.13}$$

where we define the moments of inertia with respect to the center of mass

$$\overline{\mathcal{I}}_{11} = \sum_{i=1}^{N_c} (\hat{x}_{i2}^2 + \hat{x}_{i3}^2)\, m_i, \qquad \overline{\mathcal{I}}_{22} = \sum_{i=1}^{N_c} (\hat{x}_{i1}^2 + \hat{x}_{i3}^2)\, m_i,$$
$$\overline{\mathcal{I}}_{33} = \sum_{i=1}^{N_c} (\hat{x}_{i1}^2 + \hat{x}_{i2}^2)\, m_i, \tag{5.14}$$

$$\overline{\mathcal{I}}_{12} = \overline{\mathcal{I}}_{21} = -\sum_{i=1}^{N_c} \hat{x}_{i1}\hat{x}_{i2}\, m_i, \qquad \overline{\mathcal{I}}_{23} = \overline{\mathcal{I}}_{32} = -\sum_{i=1}^{N_c} \hat{x}_{i2}\hat{x}_{i3}\, m_i,$$
$$\overline{\mathcal{I}}_{13} = \overline{\mathcal{I}}_{31} = -\sum_{i=1}^{N_c} \hat{x}_{i1}\hat{x}_{i3}\, m_i, \tag{5.15}$$

or explicitly

$$\overline{\mathcal{I}} = \begin{bmatrix} \overline{\mathcal{I}}_{11} & \overline{\mathcal{I}}_{12} & \overline{\mathcal{I}}_{13} \\ \overline{\mathcal{I}}_{21} & \overline{\mathcal{I}}_{22} & \overline{\mathcal{I}}_{23} \\ \overline{\mathcal{I}}_{31} & \overline{\mathcal{I}}_{32} & \overline{\mathcal{I}}_{33} \end{bmatrix}. \tag{5.16}$$

The particles' own inertia contribution about their respective mass-centers to the overall moment of inertia of the agglomerated body can be described by the Huygens-Steiner (generalized "parallel axis" theorem) formula ($p, s = 1, 2, 3$)

$$\bar{\mathcal{I}}_{ps} = \sum_{i=1}^{N_c} \left(\bar{\mathcal{I}}_{ps}^{i} + m_i (||\boldsymbol{r}_i - \boldsymbol{r}_{cm}||^2 \delta_{ps} - \hat{x}_{ip}\hat{x}_{is}) \right). \tag{5.17}$$

For a spherical particle, $\bar{\mathcal{I}}_{pp}^{i} = \frac{2}{5} m_i R_i^2$, and for $p \neq s$, $\bar{\mathcal{I}}_{ps}^{i} = 0$ (no products of inertia), R_i being the particle radius.[5] Finally, for the derivative of the angular momentum, utilizing $\ddot{\boldsymbol{r}}_i = \boldsymbol{a}_i = \boldsymbol{a}_{cm} + \boldsymbol{a}_{cm \to i}$, we obtain

$$\dot{\boldsymbol{H}}_{cm}^{rel} = \sum_{i=1}^{N_c} (\boldsymbol{r}_{cm\to i} \times m_i \boldsymbol{a}_{cm\to i}) = \sum_{i=1}^{N_c} (\boldsymbol{r}_{cm\to i} \times m_i (\boldsymbol{a}_i - \boldsymbol{a}_{cm})) \tag{5.18}$$

$$= \sum_{i=1}^{N_c} (m_i \boldsymbol{r}_{cm\to i} \times \boldsymbol{a}_i) \underbrace{- (\sum_{i=1}^{N_c} m_i \boldsymbol{r}_{cm\to i})}_{=0} \times \boldsymbol{a}_{cm} = \dot{\boldsymbol{H}}_{cm}, \tag{5.19}$$

and consequently

$$\dot{\boldsymbol{H}}_{cm} = \frac{d(\overline{\mathcal{I}} \cdot \boldsymbol{\omega})}{dt} = \sum_{i=1}^{N_c} \boldsymbol{r}_{cm\to i} \times \boldsymbol{\psi}_i^{ext} \stackrel{\text{def}}{=} \boldsymbol{M}_{cm}^{EXT}, \tag{5.20}$$

where $\boldsymbol{M}_{cm}^{EXT}$ is the total external moment about the center of mass.

5.2 Numerical methods for the dynamics of a DEM cluster

We now treat the dynamics of a cluster numerically. We first focus on the translational motion of the center of mass, and then turn to the rotational contribution.

5.2.1 DEM particle cluster translational contribution

The translational component of the center of mass can be written as

$$\mathcal{M} \ddot{\boldsymbol{r}}_{cm} = \mathcal{M} \dot{\boldsymbol{v}}_{cm} = \boldsymbol{\Psi}^{EXT}. \tag{5.21}$$

A trapezoidal time-stepping rule is used, whereby at some intermediate moment in time $t \leq t + \phi\Delta t \leq t + \Delta t$ ($0 \leq \phi \leq 1$)

$$\dot{\boldsymbol{v}}_{cm}(t + \phi\Delta t) \approx \frac{\boldsymbol{v}_{cm}(t + \Delta t) - \boldsymbol{v}_{cm}(t)}{\Delta t} \tag{5.22}$$

$$= \frac{1}{\mathcal{M}(t + \phi\Delta t)} \boldsymbol{\Psi}^{EXT}(t + \phi\Delta t) \tag{5.23}$$

$$\approx \frac{1}{\mathcal{M}(t + \phi\Delta t)} \left(\phi \boldsymbol{\Psi}^{EXT}(t + \Delta t) + (1 - \phi) \boldsymbol{\Psi}^{EXT}(t) \right), \tag{5.24}$$

[5] If the particles are sufficiently small, each particle's own moment inertia (about its own center) is insignificant, leading to $\bar{\mathcal{I}}_{ps} = \sum_{i=1}^{N_c} m_i (||\boldsymbol{r}_i - \boldsymbol{r}_{cm}||^2 \delta_{ps} - \hat{x}_{ip}\hat{x}_{is})$.

where $\mathcal{M}(t+\phi\Delta t) \approx \phi\mathcal{M}(t+\Delta t)+(1-\phi)\mathcal{M}(t)$, leading to

$$\boldsymbol{v}_{cm}(t+\Delta t) = \boldsymbol{v}_{cm}(t) + \frac{\Delta t}{\mathcal{M}(t+\phi\Delta t)} \left(\phi\boldsymbol{\Psi}^{EXT}(t+\Delta t) + (1-\phi)\boldsymbol{\Psi}^{EXT}(t)\right). \tag{5.25}$$

For the position, we have

$$\dot{\boldsymbol{r}}_{cm}(t+\phi\Delta t) \approx \frac{\boldsymbol{r}_{cm}(t+\Delta t)-\boldsymbol{r}_{cm}(t)}{\Delta t} \approx \boldsymbol{v}_{cm}(t+\phi\Delta t) \approx (\phi\boldsymbol{v}_{cm}(t+\Delta t) + (1-\phi)\boldsymbol{v}_{cm}(t)), \tag{5.26}$$

leading to

$$\boldsymbol{r}_{cm}(t+\Delta t) = \boldsymbol{r}_{cm}(t) + \Delta t\,(\phi\boldsymbol{v}_{cm}(t+\Delta t) + (1-\phi)\boldsymbol{v}_{cm}(t)). \tag{5.27}$$

5.2.2 DEM particle cluster rotational contribution

The quadcopter's angular velocity and rotation are determined in a similar manner by integrating the equations for an angular momentum balance

$$\dot{\boldsymbol{H}}_{cm} = \frac{d(\overline{\mathcal{I}}\cdot\boldsymbol{\omega})}{dt} = \boldsymbol{M}_{cm}^{EXT}, \tag{5.28}$$

where $\overline{\mathcal{I}}$ is the mass moment of the quadcopter, $\boldsymbol{\omega}$ is the angular velocity and $\boldsymbol{M}_{cm}^{EXT}$ is the sum of all moment contributions external to the quadcopter, around its center of mass. We remark that there are essentially two possible approaches to compute the rotational dynamics; either (1) an inertially-fixed frame or (2) a body-fixed frame. For the DEM, it is advantageous to use a inertially-fixed frame.[6] The procedure is, within a time step, to decompose an increment of motion into a rigid body translation and rotation about the center of mass. The rotation is determined by solving for the angular velocity and the subsequent incremental rotation of the body around the axis of rotation, which is aligned with the angular velocity vector direction. This leads to a coupled set of nonlinear equations which are solved iteratively.

In a fixed frame of reference the angular momentum can be written as

$$\dot{\boldsymbol{H}}_{cm} = \frac{d(\overline{\mathcal{I}}\cdot\boldsymbol{\omega})}{dt} = \boldsymbol{M}_{cm}^{EXT}. \tag{5.29}$$

$\overline{\mathcal{I}}$ is implicitly dependent on $\boldsymbol{\omega}(t)$, leading to a coupled system of nonlinear ODE's. These will be solved iteratively.

[6] For a body-fixed formulation, see Powell and Zohdi [81].

Equation 5.29 is discretized by a trapezoidal scheme

$$\frac{d(\overline{\mathcal{I}}\cdot\boldsymbol{\omega})}{dt}|_{t+\phi\Delta t} = \frac{(\overline{\mathcal{I}}\cdot\boldsymbol{\omega})|_{t+\Delta t} - (\overline{\mathcal{I}}\cdot\boldsymbol{\omega})|_t}{\Delta t}. \tag{5.30}$$

thus leading to

$$(\overline{\mathcal{I}}\cdot\boldsymbol{\omega})|_{t+\Delta t} = (\overline{\mathcal{I}}\cdot\boldsymbol{\omega})|_t + \Delta t\,\boldsymbol{M}_{cm}^{EXT}(t+\phi\Delta t). \tag{5.31}$$

Solving for $\boldsymbol{\omega}(t+\Delta t)$ yields

$$\boldsymbol{\omega}(t+\Delta t) = \left(\overline{\mathcal{I}}(t+\Delta t)\right)^{-1}\cdot\left((\overline{\mathcal{I}}\cdot\boldsymbol{\omega})|_t + \Delta t\,\boldsymbol{M}_{cm}^{EXT}(t+\phi\Delta t)\right), \tag{5.32}$$

where

$$\boldsymbol{M}_{cm}^{EXT}(t+\phi\Delta t) \approx \phi\boldsymbol{M}_{cm}^{EXT}(t+\Delta t) + (1-\phi)\boldsymbol{M}_{cm}^{EXT}(t) \tag{5.33}$$

which yields an implicit nonlinear equation, of the form $\boldsymbol{\omega}(t+\Delta t) = \mathcal{F}(\boldsymbol{\omega}(t+\Delta t))$, since $\overline{\mathcal{I}}(t+\Delta t)$, due to the body's rotation. An iterative, implicit, solution scheme may be written as follows for $K = 1, 2...$

$$\boldsymbol{\omega}^{K+1}(t+\Delta t) = \left(\overline{\mathcal{I}}^{K}(t+\Delta t)\right)^{-1}\cdot \left((\overline{\mathcal{I}}\cdot\boldsymbol{\omega})|_t + \Delta t\,\boldsymbol{M}_{cm}^{EXT,K}(t+\phi\Delta t)\right), \tag{5.34}$$

where $\overline{\mathcal{I}}^{K}(t+\Delta t)$ can be re-computed from the previous formulas. [7]. After the update for $\boldsymbol{\omega}^{K+1}(t+\Delta t)$ has been computed (utilizing the $\overline{\mathcal{I}}^{K}(t+\Delta t)$ from the previous iteration), the rotation of the body about the center of mass can be determined.

Remark 13 **-Propellor thrust** A propeller's thrust is directly proportional to its speed of rotation. This relationship is nonlinear, with airspeed, propellor design etc. controlling the overall thrust produced. In the analysis at hand, we simply assign a thrust to each motor. In the case of a hovering action, each carrying 1/4th the gravitation load. Two of the motors spin with rotation vectors pointing upwards and two pointing downwards, although the thrust is upwards for all 4 are by having the propellors flipped on two downward spinning propellors. For more details see Mueller [1].

[7] One may view the overall process as a fixed-point calculation of the form $\boldsymbol{\omega}^{K+1}(t+\Delta t) = \mathcal{F}(\boldsymbol{\omega}^{K}(t+\Delta t))$.

5.2.3 Iterative superposition scheme

The total velocity of any particle can be decomposed into the velocity of the center of mass of the entire object and the rotation of the particle relative to the center of mass:

$$\begin{aligned} \boldsymbol{v}_i &= \boldsymbol{v}_{cm} + (\boldsymbol{v}_i - \boldsymbol{v}_{cm}) = \boldsymbol{v}_{cm} + \boldsymbol{v}_{cm\to i} \\ &= \boldsymbol{v}_{cm} + \boldsymbol{\omega} \times (\boldsymbol{r}_{cm} - \boldsymbol{r}_i) = \boldsymbol{v}_{cm} + \boldsymbol{\omega} \times \boldsymbol{r}_{cm\to i} \end{aligned} \tag{5.35}$$

Explicitly, the overall motion for the bonded particles is computed by $\boldsymbol{r}_i = \boldsymbol{r}_{cm} + \boldsymbol{\omega} \times (\boldsymbol{r}_i - \boldsymbol{r}_{cm})$, sequentially by computing:

- **Velocity:** $\boldsymbol{C}_1 = \phi \boldsymbol{v}_{cm}(t + \Delta t) + (1 - \phi)\boldsymbol{v}_{cm}(t)$,
- **Angular velocity:** $\boldsymbol{C}_2 = \phi \boldsymbol{\omega}(t + \Delta t) + (1 - \phi)\boldsymbol{\omega}(t)$,
- **Center of mass position:** $\boldsymbol{C}_3 = \phi \boldsymbol{r}_{cm}(t + \Delta t) + (1 - \phi)\boldsymbol{r}_{cm}(t)$,
- **Particle positions:** $\boldsymbol{r}_i(t + \Delta t) = \boldsymbol{r}_i + \Delta t(\boldsymbol{C}_1 + \boldsymbol{C}_5)$,
- $\boldsymbol{C}_4 = \phi \boldsymbol{r}_i(t+\Delta t)+(1-\phi)\boldsymbol{r}_i(t)-\boldsymbol{C}_3$ and $\boldsymbol{C}_5 = \boldsymbol{C}_2 \times \boldsymbol{C}_4$.

5.3 Algorithmic procedure

The overall procedure is as follows, at time t:

1. Generate the quadcopter body by inserting particles within the voxel envelope/grid. Also place extra masses in the locations for the motors.
2. Set initial conditions, if $t = 0$.
3. Compute the thrust of the motors (orthogonal to the quadcopter body).
4. Compute the new position of the center of mass.
5. Compute (iteratively) the positions of the particles in the body $\boldsymbol{r}_i^K(t + \Delta t)$, $K = 1, 2, ...$:

$$||\boldsymbol{r}_i^{K+1}(t + \Delta t) - \boldsymbol{r}_i^K(t + \Delta t)|| \leq TOL||\boldsymbol{r}_i^{K+1}(t + \Delta t)||. \tag{5.36}$$

   This requires computation of the position of the center of mass, the rotation of the body, and the calculation of the positions of the particles within the iterations:
   (a) Compute/update $\boldsymbol{v}_{cm}^{K+1}(t + \Delta t) = \boldsymbol{v}_{cm}(t) + \frac{\Delta t}{\mathcal{M}(t+\phi\Delta t)} \left(\phi \boldsymbol{\Psi}^{K+1,EXT}(t + \Delta t) + (1 - \phi)\boldsymbol{\Psi}^{EXT}(t)\right)$.
   (b) Compute/update $\boldsymbol{r}_{cm}^{K+1}(t + \Delta t) = \boldsymbol{r}_{cm}(t) + \Delta t\,(\phi \boldsymbol{v}_{cm}(t + \Delta t) + (1 - \phi)\boldsymbol{v}_{cm}(t))$.
   (c) Compute/update $\boldsymbol{M}_{cm}^{EXT}(t + \phi\Delta t) \approx \phi \boldsymbol{M}_{cm}^{EXT}(t + \Delta t) + (1 - \phi)\boldsymbol{M}_{cm}^{EXT}(t)$,
   (d) Compute/update: $\boldsymbol{\omega}^{K+1}(t + \Delta t) = \left(\overline{\mathcal{I}}^K(t + \Delta t)\right)^{-1} \cdot \left((\overline{\mathcal{I}} \cdot \boldsymbol{\omega})|_t + \Delta t \boldsymbol{M}_{cm}^{EXT,K}(t + \phi\Delta t)\right)$,
   (e) Compute/update: $\boldsymbol{v}_i = \boldsymbol{v}_{cm} + \boldsymbol{\omega} \times \boldsymbol{r}_{cm\to i}$
   (f) Compute/update: $\boldsymbol{r}_i(t + \Delta t) = \boldsymbol{r}_i + \Delta t(\boldsymbol{C}_1 + \boldsymbol{C}_5)$,
   (g) Repeat steps (a)-(f) until Equation 5.36 is satisfied.
6. Increment time forward and repeat the procedure.

5.3.1 Implicit adaptive time stepping

In order to control convergence of the implicit scheme, we adopt the following algorithm:

1. **Large iteration error control:** At a time step, if after $j = 1, 2, ..., I^{upper-limit}$ internal iterations:

$$\sum_{i=1}^{N} ||\boldsymbol{r}_i^{j+1} - \boldsymbol{r}_i^j|| > TOL \Rightarrow \Delta t^{new} = \alpha \Delta t^{old}, \tag{5.37}$$

   where $0 < \alpha < 1$ and reset $j = 0$ and repeat for the time-step.
2. **Small iteration error control:** At a time step, if after $j = 1, 2, ..., I^{lower-limit}$ internal iterations:

$$\sum_{i=1}^{N} ||\boldsymbol{r}_i^{j+1} - \boldsymbol{r}_i^j|| < TOL \Rightarrow \Delta t^{new} = \beta \Delta t^{old}, \tag{5.38}$$

   where $1 < \beta < \infty$ and goto next time-step.
3. **Explicit time-stepping option:** For very small time-steps, at a time-step, after one iteration, proceed to the next time step.

Remark 14 In the current work, since there is no dislodging or breaking off of bonds, the interparticle bonds are not necessary, *in the present model.* Having the particles' motion constrained by the rigid body motion of the cluster keeps the particles bonded. In other words, no bonding laws (in the DEM sense) are required.

Remark 15 **-dislodged particles**: Although it is outside of the scope of the present work, to incorporate the possibility for particles to break off, a unilateral fragmentation threshold must be met for a particle to be deemed "dislodged", which subsequently moves according to its own dynamics. We refer the reader to Zohdi [82, 83] and [84] for details. This is discussed further in the summary and conclusions.

6 Flight-simulation of a quadcopter

(a) Basic flight modes

Starting from a basic hovering state (Figures 9 and 10), the thrusts read as

$$\boldsymbol{F}_1 + \boldsymbol{F}_2 + \boldsymbol{F}_3 + \boldsymbol{F}_4 + M\boldsymbol{g} = \boldsymbol{0} \tag{6.1}$$

and the motor torques

$$\boldsymbol{M}_1 + \boldsymbol{M}_2 + \boldsymbol{M}_3 + \boldsymbol{M}_4 = \boldsymbol{0} \tag{6.2}$$

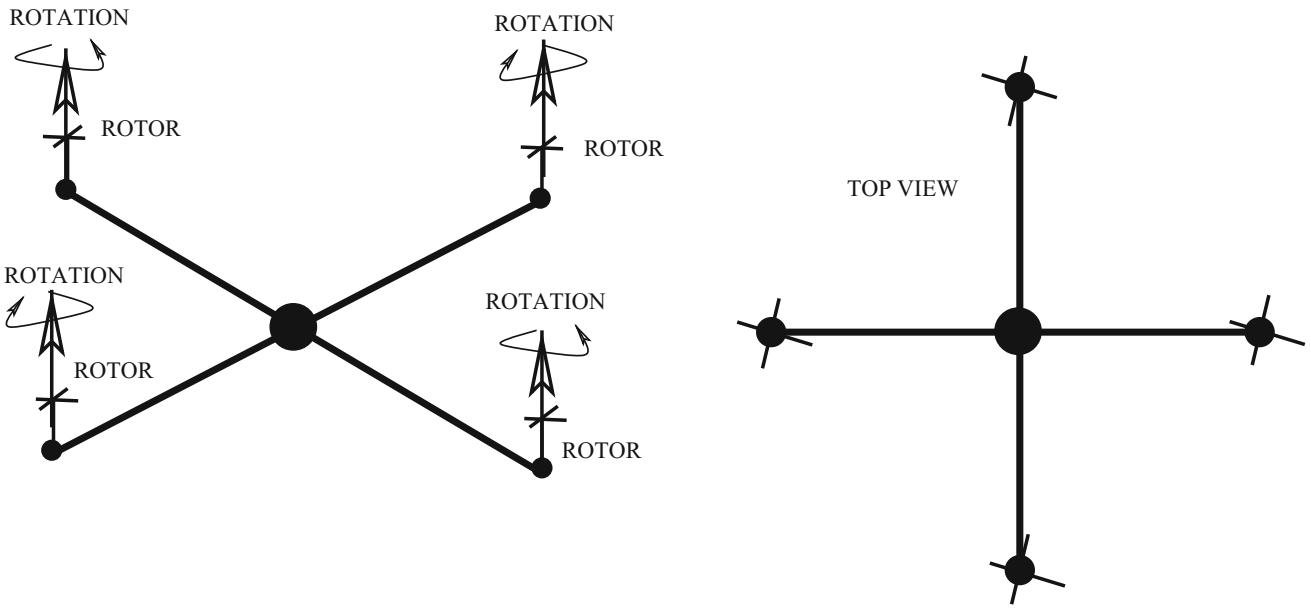

Fig. 9 A (skeletal) drone schematic with propellor thrusts. The entire body is governed by rigid body kinematics with two primary variables: the angular velocity of the body $\boldsymbol{\omega}$ and the velocity of the mass center $\boldsymbol{v}_c$. The velocities and positions of all other points on the body can be determined by rigid body kinematics. Generally, two rotors spin one way, while two of them spin the other way so that the main body of the quadcopter does not rapidly spin as it flies (conservation of angular momentum).

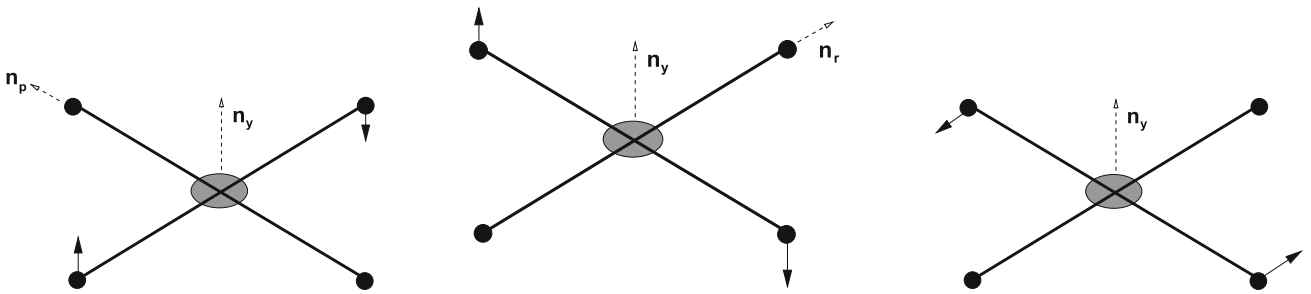

Fig. 10 A (skeletal) drone schematic with basic pitch, roll and yaw shown.

we consider three cases (1) pitch, (2) roll and (3) yaw. See the book of Mueller [1] for an in depth analysis of control paradigms.

(b) Pitch

To pitch around the $\boldsymbol{n}_p$ axis shown, one needs:

- $||\boldsymbol{F}_3|| > ||\boldsymbol{F}_1||$, which induces $||\boldsymbol{M}_3|| > ||\boldsymbol{M}_1||$,
- To suppress any unintended yaw: $\boldsymbol{M}_4$ and $\boldsymbol{M}_2$ must compensate to induce $\boldsymbol{M}_1 + \boldsymbol{M}_2 + \boldsymbol{M}_3 + \boldsymbol{M}_4 = \boldsymbol{0}$,
- To suppress any unintended drop: $(\boldsymbol{F}_1 + \boldsymbol{F}_2 + \boldsymbol{F}_3 + \boldsymbol{F}_4) \cdot \boldsymbol{n}_y - Mg = 0$.

(c) Roll

To roll around the $\boldsymbol{n}_r$ axis shown, one needs:

- $||\boldsymbol{F}_4|| > ||\boldsymbol{F}_2||$, which induces $||\boldsymbol{M}_4|| > ||\boldsymbol{M}_2||$,
- To suppress any unintended yaw: $\boldsymbol{M}_3$ and $\boldsymbol{M}_1$ must compensate to induce $\boldsymbol{M}_1 + \boldsymbol{M}_2 + \boldsymbol{M}_3 + \boldsymbol{M}_4 = \boldsymbol{0}$,
- To suppress any unintended drop: $(\boldsymbol{F}_1 + \boldsymbol{F}_2 + \boldsymbol{F}_3 + \boldsymbol{F}_4) \cdot \boldsymbol{n}_y - Mg = 0$.

(d) Yaw

To yaw around the axis $\boldsymbol{n}_y$ shown, one needs:

- $(\boldsymbol{M}_1 + \boldsymbol{M}_2 + \boldsymbol{M}_3 + \boldsymbol{M}_4) \cdot \boldsymbol{n}_y > 0$,
- To suppress any unintended drop: $(\boldsymbol{F}_1 + \boldsymbol{F}_2 + \boldsymbol{F}_3 + \boldsymbol{F}_4) \cdot \boldsymbol{n}_y - Mg = 0$.
- To suppress any unintended pitch $(\boldsymbol{M}_1^F + \boldsymbol{M}_2^F + \boldsymbol{M}_3^F + \boldsymbol{M}_4^F) \cdot \boldsymbol{n}_p = 0$, where $\boldsymbol{M}_i^F = \boldsymbol{r}_{cm \to i} \times \boldsymbol{F}_i$, is the moment induced by the thrust forces from the motors about the center of mass,
- To suppress any unintended roll $(\boldsymbol{M}_1^F + \boldsymbol{M}_2^F + \boldsymbol{M}_3^F + \boldsymbol{M}_4^F) \cdot \boldsymbol{n}_r = 0$,

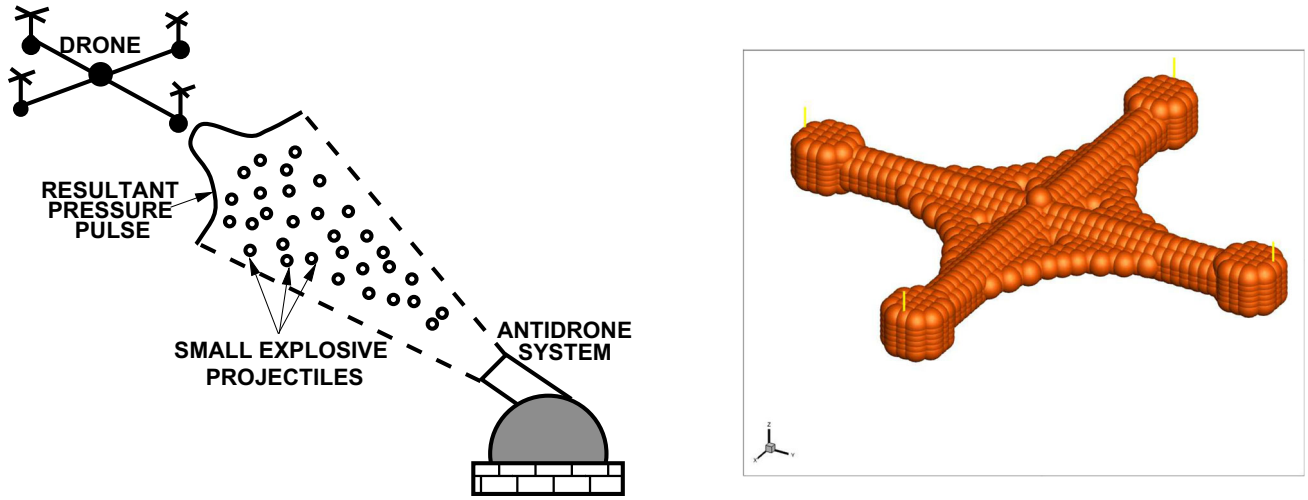

Fig. 11 Left: A drone under (destabilization) attack producing a pressure field. Right: A voxel-DEM representation

(e) Performance example: attempted destabilization of a drone

As an example, consider an antiaircraft system that explodes pressure waves at a hovering drone that is initially in yaw-mode (Figure 11) in an attempt to destabilize the it. In the example given, the equations for yaw are applied, i.e. two opposite-end motors are given larger thrusts than the other two. This induces a yaw-like motion initially. The sum of the forces are also equal and opposite to the forces of gravity ($M\boldsymbol{g}$). The forces are kept at this level for the entirety of the simulation. Specifically, we defined the thrust for each motors as

$$||\boldsymbol{F}_i|| = a_i(M^C/4 + M^m)g, \tag{6.3}$$

where

- M^C is the mass of the chassis,
- M^m is the mass of each motor,
- $a_1 = 1.25$,
- $a_2 = 0.75$,
- $a_3 = 1.25$,
- $a_4 = 0.75$,
- The proportionality constant between the motor torque and thrust: $||\boldsymbol{M}|| = \alpha||\boldsymbol{F}||$, $\alpha = 0.1$.

Both the $\boldsymbol{F}$'s and $\boldsymbol{M}$'s are projected along the normal axis of rotation of each motor. We the scenario of the drone being hit repeatedly with a set of pulses, where the maximum intensity is slightly off center (see Figures 12 and 13), with a decay that is scaled by the distance from the targeting centerline, according to:

$$P(\boldsymbol{x}) = P_o e^{-ad(\boldsymbol{x})} \tag{6.4}$$

where $P(\boldsymbol{x})$ is the pressure at $\boldsymbol{x}$, P_o is the pressure the center, $d(\boldsymbol{x})$ is the distance from the center to $\boldsymbol{x}$ and a is a decay coefficient. The following simulation parameters were chosen:

- Generation grid for DEM:$100 \times 100 \times 100$, yielding 1651 intersecting sites and hence 1651 particles,
- Total time duration:$T = 35$ seconds,
- Time step size:$\Delta t = 0.00005$ seconds,
- Starting position of center of mass: $\boldsymbol{r}_{cm}(t = 0) = \boldsymbol{0}$ (Figures 12 and 13),
- Time stepping parameter: $\phi = 0.5$ (midpoint rule),
- Drone shape exponents: $(p_1, p_2, p_3) = (0.5, 0.5, 0.5)$,
- Size of drone: $(R_1, R_2, R_3) = (0.25, 0.25, 0.05)$,
- Mass of the drone chassis: $M = 1$ kg,
- Starting angular velocity: $\boldsymbol{\omega}(t = 0) = \boldsymbol{0}$ rad/sec,
- Motor masses: $M_m = 0.25$ kg each,
- Thrust force per motor:4.55 N (this allows for perfect hovering (propellor thrust balancing gravity) if there is no external impulse),
- Density of the chassis material: $\rho = 1000\, kg/m^3$,
- Blast frequency: one every second.

As seen in Figures 12 and 13, the repeated impulses are strong enough to overturn the drone, and to destabilize it. Each entire simulation takes under a second on a standard Macbook pro. The DEM method clearly has quite a number of advantages for such an analysis, such as easy design changes, extremely fast preflight stress analysis and very fast fight analysis. However, such an analysis can also be taken a step further by formulating in-situ stress analysis in a dynamic environment. This can be achieved in two straightforward ways, namely:

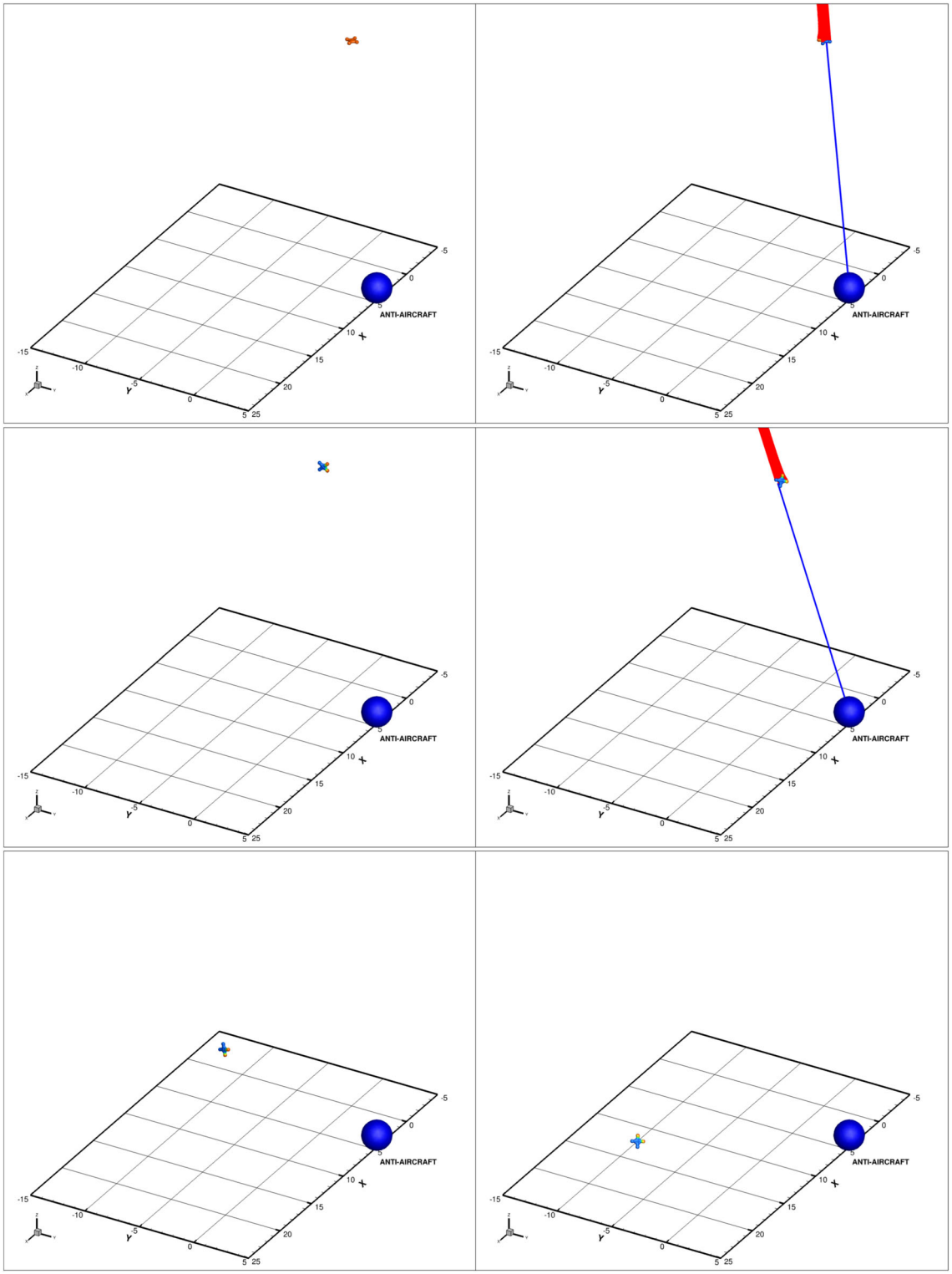

Fig. 12 Top left to right: zoom-out on initial hover and yaw then destabilization of a quadcopter by repeated explosive blasts. Orange indicates full structural integrity, while blue indicates potentially high localized structural damage (see Figure 13 for a close up view)

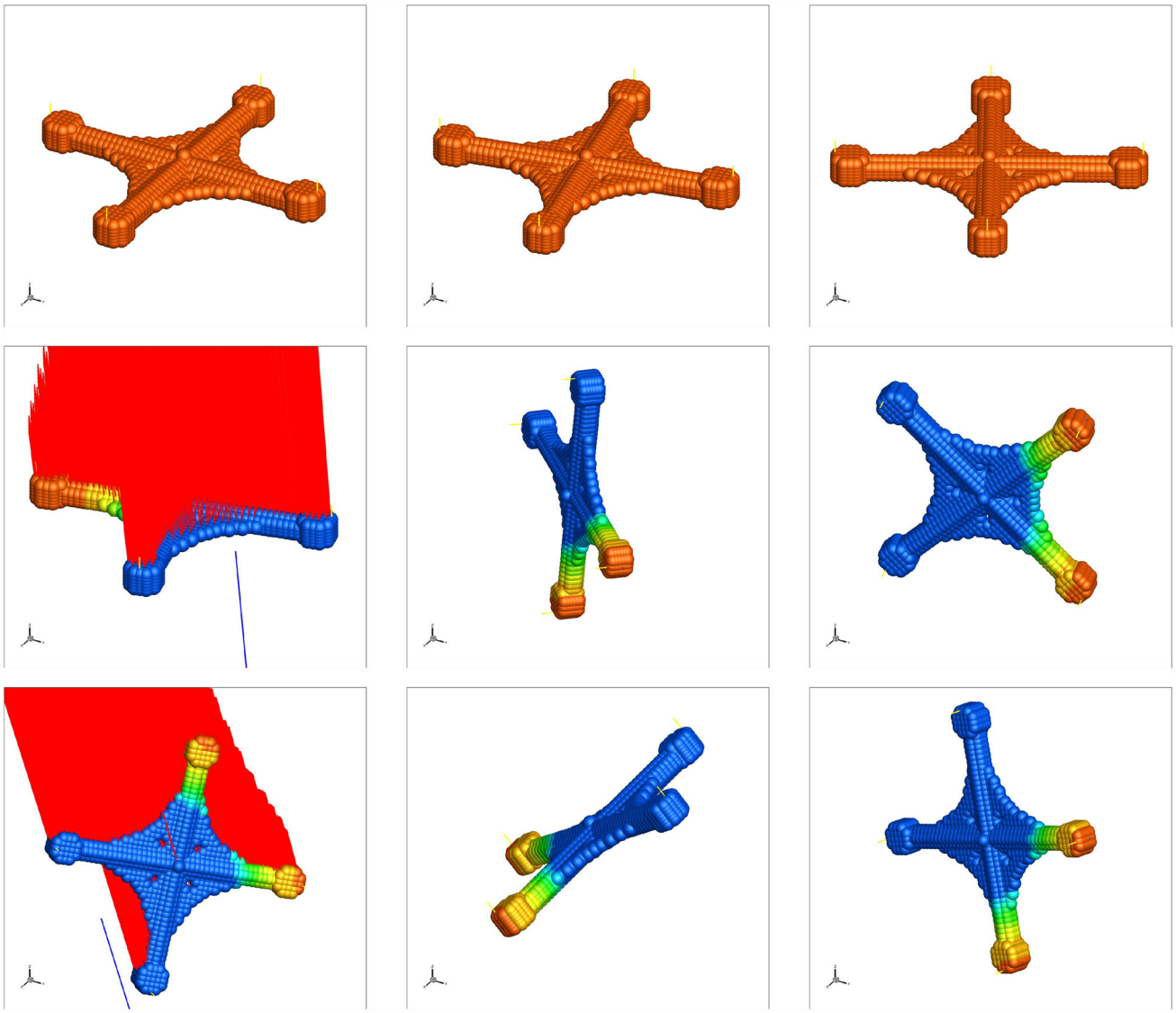

Fig. 13 Top left to right: zoom-in on initial hover and yaw then destabilization of a quadcopter by repeated explosive blasts at equal 0.1 T time intervals, where T is the total time. Orange indicates full structural integrity, while blue indicates potentially high localized structural damage (high loading)

- Method 1: Combined DEM-based Rigid-Body Dynamics with super-posed Infinitesimal deformations
- Method 2: Finite deformation DEM-based formulations

Accordingly, we discuss each method of extension.

(f) Extension method 1: post-processing rigid-motion for dynamic stress in a drone

There are a variety of methods that build upon the basic idea of computation of a rigid-motion, then the augmentation with an infinitesimal deformation analysis. There are a variety of such somewhat ad hoc approaches, which attempt to circumvent a computationally costly finite deformation analysis. One straightforward approach follows the following steps:

- Step 1: Compute the rigid motion (for example using the approach in the previous section):

$$M^{total}\frac{d\boldsymbol{v}^{rigid}}{dt} = \sum_{i=1}^{N}\boldsymbol{\Psi}_i^{ext} \tag{6.5}$$

and

$$\boldsymbol{I}^{cm}\frac{d\boldsymbol{\omega}^{rigid}}{dt} = \sum_{i=1}^{N}M_i^{cm,ext} \tag{6.6}$$

- Step 2: At each time step, compute the relative infinitesimal deformation with respect to the new rotated and

translated frame ($\boldsymbol{x}'(t)$); specifically, solve (with the decomposition of $\boldsymbol{v} = \boldsymbol{v}^{rigid} + \boldsymbol{v}^{rel}$)

$$\nabla_{x'(t)} \cdot \boldsymbol{\sigma} + \boldsymbol{f} = \rho_o \frac{d\boldsymbol{v}^{rigid}}{dt} + \rho_o \frac{\partial \boldsymbol{v}^{rel}}{\partial t}, \tag{6.7}$$

where we define

$$\boldsymbol{\psi}(\boldsymbol{x}'(t), t) \stackrel{\text{def}}{=} \frac{1}{\rho_o(\boldsymbol{x}'(t), t)} \left(\nabla_{x'(t)} \cdot \boldsymbol{\sigma} + \boldsymbol{f}(\boldsymbol{x}', t)\right) \tag{6.8}$$

where we note that

$$\frac{d\boldsymbol{v}^{rel}}{dt} = \frac{\partial \boldsymbol{v}^{rel}}{\partial t} + \nabla_{x'} \boldsymbol{v}^{rel} \cdot \underbrace{\frac{d\boldsymbol{x}'}{dt}}_{=\boldsymbol{0}} = \frac{\partial \boldsymbol{v}^{rel}}{\partial t}, \tag{6.9}$$

since it is relative to the rigid frame.

- Step 3: Compute the relative velocity

$$\boldsymbol{v}^{rel}(\boldsymbol{x}', t + \Delta t) \approx \boldsymbol{v}^{rel}(\boldsymbol{x}', t) + \Delta t \left(\phi \boldsymbol{\psi}(\boldsymbol{x}', t + \Delta t) + (1 - \phi)\boldsymbol{\psi}(\boldsymbol{x}', t)\right) \tag{6.10}$$

- Step 4: Compute the position of each voxel, relative to the moving frame:

$$\boldsymbol{x}^{rel}(\boldsymbol{x}', t + \Delta t) \approx \boldsymbol{x}^{rel}(\boldsymbol{x}', t) + \Delta t \left(\phi \boldsymbol{v}^{rel}(\boldsymbol{x}', t + \Delta t) + (1 - \phi)\boldsymbol{v}^{rel}(\boldsymbol{x}', t)\right). \tag{6.11}$$

Such approaches are rather popular in flight dynamics, however, the most rigorous approach is to formulate a fully finite-deformation structural dynamics problem. This is discussed next.

(g) Extension method 2: fully finite-deformation structural dynamics

The extension of the preceding procedures to finite deformation problems (finite strains and large displacements and rotations) is rather straightforward and can be achieved by performing the presented calculations for the field equations specifically in the reference configuration ($\boldsymbol{x} \neq \boldsymbol{X}$)

$$\nabla_X \cdot \boldsymbol{P} + \boldsymbol{f} J = \rho_0 \frac{d\boldsymbol{v}}{dt} \quad \text{or} \quad \nabla_X \cdot (\boldsymbol{F} \cdot \boldsymbol{S}) + \boldsymbol{f} J = \rho_0 \frac{d\boldsymbol{v}}{dt}, \tag{6.12}$$

where $\boldsymbol{F} = \nabla_X \boldsymbol{x}$ is the Deformation Gradient, $J = \det \boldsymbol{F}$ is the Jacobian, $\boldsymbol{P} = \boldsymbol{\sigma} \cdot \boldsymbol{F}^{-T} (\det \boldsymbol{F})$ is the First Piola-Kirchhoff stress and $\boldsymbol{S} = J \boldsymbol{F}^{-1} \cdot \boldsymbol{\sigma} \cdot \boldsymbol{F}^{-T}$ is the Second Piola-Kichhoff stress. Specifically, the algorithm is, using the voxel-stencils in the reference configuration (with referential $X's$ replacing $x's$ in the voxel approximations for derivatives, gradients, etc.):

- Step 1: Define

$$\boldsymbol{\Psi}(\boldsymbol{X}, t) \stackrel{\text{def}}{=} \frac{1}{\rho_o(\boldsymbol{X}, t)} \left(\nabla_X \cdot (\boldsymbol{F}(\boldsymbol{X}, t) \cdot \boldsymbol{S}(\boldsymbol{X}, t)) + \boldsymbol{f}(\boldsymbol{X}, t) J(\boldsymbol{X}, t)\right) \tag{6.13}$$

where we note that

$$\frac{d\boldsymbol{v}}{dt} = \frac{\partial \boldsymbol{v}}{\partial t} + \nabla_X \boldsymbol{v} \cdot \underbrace{\frac{d\boldsymbol{X}}{dt}}_{=\boldsymbol{0}} = \frac{\partial \boldsymbol{v}}{\partial t} \tag{6.14}$$

- Step 2: Compute the velocity

$$\boldsymbol{v}(\boldsymbol{X}, t + \Delta t) \approx \boldsymbol{v}(\boldsymbol{X}, t) + \Delta t \left(\phi \boldsymbol{\Psi}(\boldsymbol{X}, t + \Delta t) + (1 - \phi)\boldsymbol{\Psi}(\boldsymbol{X}, t)\right) \tag{6.15}$$

- Step 3: Compute the position of each voxel (originally in the reference configuration $\boldsymbol{X}$)

$$\boldsymbol{x}(\boldsymbol{X}, t + \Delta t) \approx \boldsymbol{x}(\boldsymbol{X}, t) + \Delta t \left(\phi \boldsymbol{v}(\boldsymbol{X}, t + \Delta t) + (1 - \phi)\boldsymbol{v}(\boldsymbol{X}, t)\right). \tag{6.16}$$

There can be a variety of reason that may motivate this, such as, for example, large-scale damage and fragmentation, which we briefly discuss next.

(h) Extensions to include fragmentation

Importantly, although we have not discussed the evolution of damage (see Zohdi [28] for coarse ad-hoc heuristics), these approaches are important for ascertaining whether a discrete element will become dislodged by checking a unilateral *integrity bonding criteria* between particles:

- Remains bonded/intact: $||\boldsymbol{\Psi}(\boldsymbol{x})|| \leq \Psi^* \Rightarrow \dot{\gamma}(\boldsymbol{x}) = 0$.
- Debonding evolution: $||\boldsymbol{\Psi}(\boldsymbol{x})|| \geq \Psi^* \Rightarrow \dot{\gamma}(\boldsymbol{x}) = \alpha(1 - \frac{||\boldsymbol{\Psi}||}{\Psi^*})$,

where α is a rate parameter and γ is a integrity parameter, where initially $\gamma(\boldsymbol{x}, t = 1) = 1$; i.e. no damage/full integrity. When $\gamma \leq TOL$ then the particle is allowed to debond and mass is lost. We define γ as the integrity and $1 - \gamma$ as damage. If the part of the body separates then it is released with the position and velocity of the body at that point. The contact mechanics of dislodged particles with other dislodged material, the remainder of the drone hull, etc., is outside the scope of the present work. As an example, consider a debonding relation that is dependent on the intensity of the external blast force applied:

- Remains bonded/intact: $||\boldsymbol{\Psi}^{ext}(\boldsymbol{x})|| \leq \Psi^* \Rightarrow \dot{\gamma}(\boldsymbol{x}) = 0$,
- Progressive damage/debonding: $||\boldsymbol{\Psi}^{ext}(\boldsymbol{x})|| \geq \Psi^* \Rightarrow \dot{\gamma}(\boldsymbol{x}) = \alpha(1 - \frac{||\boldsymbol{\Psi}^{ext}||}{\Psi^*})$

One important feature of the DEM methods is that one can incorporate the dislodging of the particles, for example when the following criteria is met, $\gamma(\boldsymbol{x}) \leq TOL$, then the fragment is released. Clearly, the interaction of clusters of fragments, with a full contact analysis, is a logical extension of the analysis presented. This would require a direct numerical simulation-type formulation for dynamics of a multi-particulate system ($i = 1, 2...N$ bonded and fragmented particles)

$$m_i \ddot{\boldsymbol{r}}_i = \boldsymbol{\Psi}_i^{tot}(\boldsymbol{r}_1, \boldsymbol{r}_2, ..., \boldsymbol{r}_{N_p}, \text{contact forces, bonding forces}) \tag{6.17}$$

where $\boldsymbol{r}_i$ is the position vector of the ith particle, $\boldsymbol{\Psi}_i^{tot}$ represents all forces acting on particle i, with contact detection and contact force computation. In Zohdi [28], simplified versions of this fragmentation extension were implemented, and we refer the reader to that work. The simulation of related flowing particulate/fragmented systems has been extensively investigated for the last decade by Zohdi [82, 83], employing numerical schemes based on high-performance iterative solvers, sorting-binning for fast inter-particle calculations, Verlet lists, domain decomposition, parallel processing, etc. These types of formulations are able to characterize the interaction of multiple fragments from breakup/disintegration and blasts where the application of purely continuum approaches would be virtually impossible. The dynamics of fragments of clusters that evolve and interact with the quadcopter and other fragments is complex, and algorithmically aligns closely to the study of "granular" or "particulate" media, for example see Duran [85], Pöschel and Schwager [86], Onate et al. [87, 88], Rojek et al. [89], Carbonell et al. [90], Labra and Onate [91] and Zohdi [82, 83]. The application of extension method 1 (small deformations super posed on large displacements), extension method 2 (finite deformation formulation) and fragmentation is under investigation by the author.

7 Summary and extensions

The presented work is geared towards accelerating the development of drones for a wide-range of applications. The main components are (1) modular assembly of structural components and (2) flexible feature addition and deletion and dynamic drone flight performance evaluation. As a model problem, we illustrated the concepts above for the preflight stress testing of a quadcopter and its subsequent dynamic flight performance in an extreme environment. Numerical examples were provided to illustrate the overall approach and its straightforward implementation. The further development of such tools, such as finite deformation formulations and fragmentation, are currently under investigation by the author. It is important to realize the role of drones in our rapidly evolving world. The overall worldwide rise of AI and machine-learning have spurred on large-scale investments focussed on the development of multi-drone-swarm technologies. A central goal is the development of machine-learning enabled digital-twins to rapidly ascertain optimal programming for desired tactical multi-drone swarm like behavior. Swarm modeling has origins in the description of biological groups (flocks of birds, schools of fish, crowds of human beings, etc.), as. well as predator-prey relationships (Breder [92], 1952). Early approaches that rely on decentralized organization can be found in Beni [93], Brooks [94], Dude et al [95], Cao et al [96], Liu and Passino [97] and Turpin et al [98]. Usual models incorporate a tradeoff between long-range interaction and short-range repulsion between individuals, dependent on the relative distance between individuals. The most basic model is to treat each individual as a point mass and to allow the overall multi-body system to dynamically move in response to its environment, based on Newtonian mechanics (Gazi and Passino [99], Bender and Fenton [100], Kennedy and Eberhart [101], Goodrich et al. [105] and Zohdi [28, 102–104]).[8] For some creatures, the "visual field" of individuals may play a significant role, while if the agents are robots or drones, the communication can be electronic.[9]

In order to achieve this type of high-performance, typically, such drones require a variety of sensors on board, ranging from 6 DOF 3-axis gyroscopes, a array of accelerometers, Inertial Measurement Units (IMU's), barometers and GPS receivers. They also have a number of actuators, such as digital electronic speed controllers that are connected to engine, motors, servomotors and propellors. There are numerous emerging technologies that have been enabled by drones, driven primarily by defense systems. As a result, there are consequential geopolitical issues associated with *quasi-commericial* drones, in particular related to security issues. While small-scale commercial drones were initially considered of little concern, recent wars, such as the Ukraine

[8] There are other modeling paradigms, for example mimicing ant colonies (Bonabeau et al., [106]), which exhibit foraging-type behavior and trail-laying-trail-following mechanisms for finding food sources (see Kennedy and Eberhart [101] and Bonabeau et. al [106], Dorigo et. al, [107], Bonabeau et. al [106], Bonabeau and Meyer [108] and Fiorelli et al [109]).

[9] However, in some systems, agents interact with a specific set of other agents, *regardless* of whether they are far away (Feder [110]). This appears to be the case for Starlings (Sturnus vulgaris), where Ballerini et al [111] concluded, that such birds communicate with a certain number of birds surrounding it and that that interactions are governed by topological distance and not metric distance.

conflict, have starkly altered that view. In particular, First-Person Viewer (FPV) drones, which stream a video to a ground-based controller, have become centrally important. As mentioned at the beginning of this work, several key components common to most drones: (1) Lightweight airframes, (2) Brushless DC motors, (3) Power transmission, (4) Flight controls, (5) Batteries and power supply, (6) Cameras, (7) Sensors, (8) Navigation, (9) Onboard computing, processing and memory and (10) Wireless communication. Additionally, there are highly specialized materials involved, such as sintered permanent neodymium (NdFeB) for brushless DC motors and lithium polymers (LiPo) for lightweight batteries are critical components. In particular, over the last 30 years, battery technology utilize gel polymers and solid-state electrolytes to deliver extremely high energy per unit of weight that have led to approximately a 300 % increase in energy density and nearly a 100 % reduction in cost of lithium-ion batteries. As of 2025, companies based in China dominate the commercial drone market, with overall estimates of nearly 85 % market share (see popular press articles [112–132]), producing drones that are easy to use, simple to maintain and relatively inexpensive, backed by a skilled workforce who is able to supply and innovate products at unmatched rapid speeds. However, as mentioned previously, over that last two years, there has been huge worldwide investments, in particular by the US, Taiwan, Europe in drone technologies, in parallel with the rise of AI and machine-learning, which has spurred on research focussing on collaborative multi-drone swarm technologies. In this regard, as mentioned, advances in camera technologies have become critical for drone-swarm technologies to develop further, which is a current focus of the author.

8 Appendix 1: more on property estimates

There are a number of improvements and extensions possible for effective property estimates. For example, a critical observation is that the lower Hashin-Shtrikman bound is more accurate when the material is composed of stiff particles that are surrounded by a soft matrix (denoted case 1) and the upper bound is more accurate for a stiff matrix surrounding soft particles (denoted case 2). For example, this can be explained by considering two cases of material combinations, one with with 50 % soft material and 50 % stiff material. A material with a continuous soft binder (50 %) will isolate the stiff particles (50 %), and the overall system will not be stiff (this is case 1 and the lower bound is more accurate), while a material formed by a continuous stiff binder (50 %) surrounding soft particles (50 %, case 2) has an overall stiffness higher than case 1. Thus, case 2 is more closely approximated by the upper bound and case 1 is closer to the lower bound. ϕ *is function of the microstructure, and must be calibrated.* As mentioned, for stiff spherical particles, at low volume fractions, for example under 15 %, where the particles are not in contact, the lower bound is more accurate. Thus, one would pick $\phi \leq 0.5$ to bias the estimate to the lower bound. However, if we take the same volume fraction of particles, but use flat flakes or chopped fibers, they will certainly touch, and produce stiff pathways (Figure 14). Their overall stiffness will certainly be higher than those of sphere at the same volume fraction. Thus, one would pick $\phi \geq 0.5$. One can calibrate ϕ by comparing it to different experiments (see, for example, Zohdi et al [67]). Essentially, the more the particles interact, for example physically touch, the more the upper bound becomes relevant. The general trends are (a) for cases where the upper bound is more accurate, $\phi \geq 0.5$ and (b) for cases when the lower bound is more accurate, $\phi \leq 0.5$. ϕ reflects the degree of interaction of the particulate constituents. Also, in order to make estimates of the load-sharing for a mixture of more than two materials, one can first use a multiphase extension of the Hashin-Shtrikman bounds (Hashin and Shtrikman [80]) and then follow a similar procedure as that found in this work to construct multiphase concentration functions. This is done in Appendix 2.

Remark 16 Although not directly needed in the upcoming analysis, for the strain, we have for the matrix,

$$\begin{aligned}\langle \boldsymbol{\epsilon} \rangle_{\Omega_1} &= \frac{\langle \boldsymbol{\epsilon} \rangle_{\Omega} - v_2 \langle \boldsymbol{\epsilon} \rangle_{\Omega_2}}{v_1} = \frac{\langle \boldsymbol{\epsilon} \rangle_{\Omega} - v_2 \boldsymbol{C}^{\epsilon,2} : \langle \boldsymbol{\epsilon} \rangle_{\Omega}}{v_1} \\ &= \frac{(\mathbf{1} - v_2 \boldsymbol{C}^{\epsilon,2}) : \langle \boldsymbol{\epsilon} \rangle_{\Omega}}{v_1} \stackrel{\text{def}}{=} \boldsymbol{C}^{\epsilon,1} : \langle \boldsymbol{\epsilon} \rangle_{\Omega}.\end{aligned} \tag{8.1}$$

Therefore, in the case of isotropy,

$$C_{\kappa}^{\epsilon,1} \stackrel{\text{def}}{=} \frac{1}{v_1}(1 - v_2 C_{\kappa}^{\epsilon,2}) \qquad \text{and} \qquad C_{\mu}^{\epsilon,1} \stackrel{\text{def}}{=} \frac{1}{v_1}(1 - v_2 C_{\mu}^{\epsilon,2}). \tag{8.2}$$

As for the stress, the portion of the total strain carried by each phase can be determined by multiplying the concentration factors by the corresponding volume fractions

$$\begin{aligned}&\langle \boldsymbol{\epsilon} \rangle_{\Omega} = v_1 \langle \boldsymbol{\epsilon} \rangle_{\Omega_1} + v_2 \langle \boldsymbol{\epsilon} \rangle_{\Omega_2} = v_1 \boldsymbol{C}^{\epsilon,1} : \langle \boldsymbol{\epsilon} \rangle_{\Omega} + v_2 \boldsymbol{C}^{\epsilon,2} : \\ &\langle \boldsymbol{\epsilon} \rangle_{\Omega} = (v_1 \boldsymbol{C}^{\epsilon,1} + v_2 \boldsymbol{C}^{\epsilon,2}) : \langle \boldsymbol{\epsilon} \rangle_{\Omega}.\end{aligned} \tag{8.3}$$

In the case of isotropy

$$\langle \boldsymbol{\epsilon} \rangle_{\Omega} = (v_1 C^{\epsilon,1} + v_2 C^{\epsilon,2}) \langle \boldsymbol{\epsilon} \rangle_{\Omega} \Rightarrow v_1 C^{\epsilon,1} + v_2 C^{\epsilon,2} = 1. \tag{8.4}$$

9 Appendix 2: multi-component material systems

In order to make estimates of the load-share levels for multiphase materials of a mixture of more than two materials, we again employ multicomponent Hashin-Shtrikman bounds (Hashin and Shtrikman [80]).

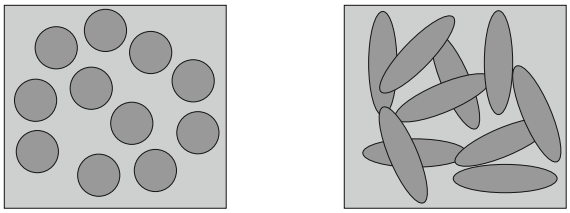

Fig. 14 "Comparing microstructures with the same volume fractions. Nonspherical touch more, and thus need a higher value of ϕ

9.1 Bulk modulus

Consider a material with N different phases, κ_1, $\kappa_2 ... \kappa_N$, where κ_N has the highest bulk modulus and κ_1, has the lowest. The corresponding volume fractions are v_1, $v_2 ... v_N$. The Hashin-Shtrikman bounds are

$$\kappa^{*,-} = \kappa_1 + \frac{A_1}{1 - \alpha_1 A_1} \leq \kappa^* \leq \kappa_N + \frac{A_N}{1 - \alpha_N A_N} \kappa^{*,+}, \quad (9.1)$$

where

$$\alpha_1 = \frac{3}{3\kappa_1 + 4\mu_1}, \quad (9.2)$$

and

$$\alpha_N = \frac{3}{3\kappa_N + 4\mu_N}, \quad (9.3)$$

and

$$A_1 = \sum_{i=2}^{N} \frac{v_i}{(\kappa_i - \kappa_1)^{-1} + \alpha_1} \quad (9.4)$$

and

$$A_N = \sum_{i=1}^{N-1} \frac{v_i}{(\kappa_i - \kappa_N)^{-1} + \alpha_N} \quad (9.5)$$

For a two phase microstructure, this collapses to the classical bounds

$$\underbrace{\kappa_1 + \frac{v_2}{\frac{1}{\kappa_2 - \kappa_1} + \frac{3(1-v_2)}{3\kappa_1 + 4\mu_1}}}_{\text{bulk modulus H}-\text{S lower bound} \stackrel{\text{def}}{=} \kappa^{*,-}} \leq \kappa^*$$

$$\leq \underbrace{\kappa_2 + \frac{1 - v_2}{\frac{1}{\kappa_1 - \kappa_2} + \frac{3v_2}{3\kappa_2 + 4\mu_2}}}_{\text{bulk modulus H}-\text{S upper bound} \stackrel{\text{def}}{=} \kappa^{*,+}}, \quad (9.6)$$

where κ_1, μ_1 and κ_2, μ_2 are the bulk and shear moduli for the phases, while v_2 is the phase 2 volume fraction.

9.2 Shear modulus

Consider a material with N different phases, μ_1, $\mu_2 ... \mu_N$, where μ_N has the highest shear modulus and μ_1, has the lowest. The corresponding volume fractions are v_1, $v_2 ... v_N$. The Hashin-Shtrikman bounds are

$$\mu^{*,-} = \mu_1 + \frac{B_1}{1 - \gamma_1 B_1} \leq \mu^* \leq \mu_N + \frac{B_N}{1 - \gamma_N B_N} \mu^{*,+}, \quad (9.7)$$

where

$$\gamma_1 = \frac{3(\kappa_1 + \mu_1)}{5\mu_1(3\kappa_1 + 4\mu_1)}, \quad (9.8)$$

and

$$\gamma_N = \frac{3(\kappa_N + \mu_N)}{5\mu_N(3\kappa_N + 4\mu_N)}, \quad (9.9)$$

and

$$B_1 = \sum_{i=2}^{N} \frac{v_i}{(\mu_i - \mu_1)^{-1} + \gamma_1} \quad (9.10)$$

and

$$B_N = \sum_{i=1}^{N-1} \frac{v_i}{(\mu_i - \mu_N)^{-1} + \gamma_N} \quad (9.11)$$

For a two phase microstructure, this collapses to

$$\underbrace{\mu_1 + \frac{v_2}{\frac{1}{\mu_2 - \mu_1} + \frac{6(1-v_2)(\kappa_1+2\mu_1)}{5\mu_1(3\kappa_1+4\mu_1)}}}_{\text{shear modulus H}-\text{S lower bound} \stackrel{\text{def}}{=} \mu^{*,-}} \leq \mu^*$$

$$\leq \underbrace{\mu_2 + \frac{(1 - v_2)}{\frac{1}{\mu_1 - \mu_2} + \frac{6v_2(\kappa_2+2\mu_2)}{5\mu_2(3\kappa_2+4\mu_2)}}}_{\text{shear modulus H}-\text{S upper bound} \stackrel{\text{def}}{=} \mu^{*,+}}. \quad (9.12)$$

9.3 Concentration tensors for multiphase materials

As for a two-phase material, the load carried by each phase in the microstructure is characterized via stress and strain concentration tensors. One can decompose the stress average Ω into averages over the each of the phases in the following manner, recall:

$$\begin{aligned}\langle\boldsymbol{\sigma}\rangle_{\Omega} &= \frac{1}{|\Omega|}\left(\int_{\Omega_1}\boldsymbol{\sigma}\,d\Omega+\int_{\Omega_2}\boldsymbol{\sigma}\,d\Omega+\ldots+\int_{\Omega_N}\boldsymbol{\sigma}\,d\Omega\right)\\ &= v_1\langle\boldsymbol{\sigma}\rangle_{\Omega_1}+v_2\langle\boldsymbol{\sigma}\rangle_{\Omega_2}+\ldots=\sum_{i=1}^{N}v_i\langle\boldsymbol{\sigma}\rangle_{\Omega_i}.\end{aligned} \tag{9.13}$$

If we make use of this decomposition, we have

$$\begin{aligned}\langle\boldsymbol{\sigma}\rangle_{\Omega} &= \sum_{i=1}^{N}v_i\langle\boldsymbol{\sigma}\rangle_{\Omega_i}=\sum_{i=1}^{N}v_i\boldsymbol{I\!E}_i:\langle\boldsymbol{\epsilon}\rangle_{\Omega_i}\\ &= \boldsymbol{I\!E}_1:(\langle\boldsymbol{\epsilon}\rangle_{\Omega}-\sum_{j=2}^{N}v_2\langle\boldsymbol{\epsilon}\rangle_{\Omega_j})+\sum_{j=2}^{N}v_j\boldsymbol{I\!E}_j:\langle\boldsymbol{\epsilon}\rangle_{\Omega_j}\\ &= \left(\boldsymbol{I\!E}_1+\sum_{j=2}^{N}v_2(\boldsymbol{I\!E}_j-\boldsymbol{I\!E}_1):\boldsymbol{C}^{\epsilon,j}\right):\langle\boldsymbol{\epsilon}\rangle_{\Omega}\end{aligned} \tag{9.14}$$

where $\boldsymbol{C}^{\epsilon,j}\overset{\text{def}}{=}\left(\frac{1}{v_j}(\boldsymbol{I\!E}_j-\boldsymbol{I\!E}_1)^{-1}:(\boldsymbol{I\!E}^*-\boldsymbol{I\!E}_j)\right)$ with $\boldsymbol{C}^{\epsilon,j}:\langle\boldsymbol{\epsilon}\rangle_{\Omega}=\langle\boldsymbol{\epsilon}\rangle_{\Omega_j}$. The strain concentration tensor $\boldsymbol{C}^{\epsilon,j}$ relates the average strain over the phase (j) to the average strain over all phases. Similarly, for the variation in the stress we have $\boldsymbol{C}^{\epsilon,j}:\boldsymbol{I\!E}^{*-1}:\langle\boldsymbol{\sigma}\rangle_{\Omega}=\boldsymbol{I\!E}_j^{-1}:\langle\boldsymbol{\sigma}\rangle_{\Omega_j}$, which reduces to $\boldsymbol{I\!E}_j:\boldsymbol{C}^{\epsilon,j}:\boldsymbol{I\!E}^{*-1}:\langle\boldsymbol{\sigma}\rangle_{\Omega}\overset{\text{def}}{=}\boldsymbol{C}^{\sigma,j}:\langle\boldsymbol{\sigma}\rangle_{\Omega}=\langle\boldsymbol{\sigma}\rangle_{\Omega_j}$. $\boldsymbol{C}^{\sigma,j}$ is known as the stress concentration tensor; it relates the average stress in the particle phase to that in the whole RVE. Note that once either the $\boldsymbol{C}^{\sigma,j}$'s or $\boldsymbol{I\!E}^*$ are known, the other can be determined. In the case of isotropy we may write ($j\neq 1$)

$$C_{\kappa}^{\sigma,j}\overset{\text{def}}{=}\frac{1}{v_2}\frac{\kappa_j}{\kappa^*}\frac{\kappa^*-\kappa_1}{\kappa_j-\kappa_1}\qquad\text{and}\qquad C_{\mu}^{\sigma,j}\overset{\text{def}}{=}\frac{1}{v_j}\frac{\mu_j}{\mu^*}\frac{\mu^*-\mu_1}{\mu_j-\mu_1} \tag{9.15}$$

where $C_{\kappa}^{\sigma,j}\langle\frac{tr\boldsymbol{\sigma}}{3}\rangle_{\Omega}=\langle\frac{tr\boldsymbol{\sigma}}{3}\rangle_{\Omega_j}$ and where $C_{\mu}^{\sigma,j}\langle\boldsymbol{\sigma}'\rangle_{\Omega}=\langle\boldsymbol{\sigma}'\rangle_{\Omega_j}$. Clearly, the microstress fields are minimally distorted when $C_{\kappa}^{\sigma,j}=C_{j,\mu}^{\sigma,2}=1$; there are no stress concentrations in a homogeneous material. For the matrix,

$$\begin{aligned}\langle\boldsymbol{\sigma}\rangle_{\Omega_1} &= \frac{\langle\boldsymbol{\sigma}\rangle_{\Omega}-\sum_{j=2}^{N}v_j\langle\boldsymbol{\sigma}\rangle_{\Omega_j}}{v_1}\\ &= \frac{\langle\boldsymbol{\sigma}\rangle_{\Omega}-\sum_{j=2}^{N}v_j\boldsymbol{C}^{\sigma,j}:\langle\boldsymbol{\sigma}\rangle_{\Omega}}{v_1}\\ &= \frac{(\mathbf{1}-\sum_{j=2}^{N}v_j\boldsymbol{C}^{\sigma,j}):\langle\boldsymbol{\sigma}\rangle_{\Omega}}{v_1}\overset{\text{def}}{=}\boldsymbol{C}^{\sigma,1}:\langle\boldsymbol{\sigma}\rangle_{\Omega}.\end{aligned} \tag{9.16}$$

Therefore, in the case of isotropy,

$$\begin{aligned}C_{\kappa}^{\sigma,1}&\overset{\text{def}}{=}\frac{1}{v_1}(1-\sum_{j=2}^{N}v_jC_{\kappa}^{\sigma,j})\qquad\text{and}\\ C_{\mu}^{\sigma,1}&\overset{\text{def}}{=}\frac{1}{v_1}(1-\sum_{j=2}^{N}v_jC_{\mu}^{\sigma,j}).\end{aligned} \tag{9.17}$$

The portion of the total stress carried by each phase can be determined by multiplying the concentration factors by the corresponding volume fractions

$$\begin{aligned}\langle\boldsymbol{\sigma}\rangle_{\Omega} &= v_1\langle\boldsymbol{\sigma}\rangle_{\Omega_1}+\sum_{j=2}^{N}v_j\langle\boldsymbol{\sigma}\rangle_{\Omega_j}=v_1\boldsymbol{C}^{\sigma,1}:\langle\boldsymbol{\sigma}\rangle_{\Omega}\\ &\quad+\sum_{j=2}^{N}v_j\boldsymbol{C}^{\sigma,j}:\langle\boldsymbol{\sigma}\rangle_{\Omega}.\end{aligned} \tag{9.18}$$

10 Appendix 3: Recursion formula for second-order temporal discretization

Discretization of temporally second-order equations can be illustrated by considering

$$\ddot{\boldsymbol{U}}=\dot{\boldsymbol{V}}=\boldsymbol{L}(t). \tag{10.1}$$

where $\boldsymbol{U}$ is the vector of all voxel displacement values and $\boldsymbol{V}$ is the vector of all voxel velocity values. Explicitly, $\boldsymbol{U}$ is the vector of all voxel values in the system

$$\boldsymbol{U}\overset{\text{def}}{=}\left\{\begin{array}{c}U_1\\U_2\\U_3\\\ldots\\U_N\end{array}\right\} \tag{10.2}$$

and

$$\boldsymbol{V} \stackrel{\text{def}}{=} \left\{ \begin{array}{c} V_1 \\ V_2 \\ V_3 \\ \ldots \\ V_N \end{array} \right\} \tag{10.3}$$

and

$$\boldsymbol{L} \stackrel{\text{def}}{=} \left\{ \begin{array}{c} L_1 \\ L_2 \\ L_3 \\ \ldots \\ L_N \end{array} \right\} \tag{10.4}$$

where $\boldsymbol{L}$ represents all of the nodal values of $\boldsymbol{\Psi}$ found in Equation 3.4. Expanding the field $\boldsymbol{V}$ in a Taylor series about $t + \phi\Delta t$ we obtain

$$\boldsymbol{V}(t+\Delta t) \approx \boldsymbol{V}(t+\phi\Delta t) + \frac{d\boldsymbol{V}}{dt}|_{t+\phi\Delta t}(1-\phi)\Delta t + \frac{1}{2}\frac{d^2\boldsymbol{V}}{dt^2}|_{t+\phi\Delta t}(1-\phi)^2(\Delta t)^2 + \mathcal{O}((\Delta t)^3) \tag{10.5}$$

and

$$\boldsymbol{V}(t) \approx \boldsymbol{V}(t+\phi\Delta t) - \frac{d\boldsymbol{V}}{dt}|_{t+\phi\Delta t}\phi\Delta t + \frac{1}{2}\frac{d^2\boldsymbol{V}}{dt^2}|_{t+\phi\Delta t}\phi^2(\Delta t)^2 + \mathcal{O}((\Delta t)^3) \tag{10.6}$$

Subtracting the two expressions yields

$$\frac{d\boldsymbol{V}}{dt}|_{t+\phi\Delta t} \approx \frac{\boldsymbol{V}(t+\Delta t) - \boldsymbol{V}(t)}{\Delta t} + \hat{\mathcal{O}}(\Delta t), \tag{10.7}$$

where $\hat{\mathcal{O}}(\Delta t) = \mathcal{O}((\Delta t)^2)$, when $\phi = \frac{1}{2}$. Thus, inserting this into the governing equation yields

$$\boldsymbol{V}(t+\Delta t) \approx \boldsymbol{V}(t) + \Delta t\boldsymbol{L}(t+\phi\Delta t) + \hat{\mathcal{O}}((\Delta t)^2). \tag{10.8}$$

Note that adding a weighted sum of Equations 10.5 and 10.6 yields

$$\boldsymbol{V}(t+\phi\Delta t) \approx \phi\boldsymbol{V}(t+\Delta t) + (1-\phi)\boldsymbol{V}(t) + \mathcal{O}((\Delta t)^2), \tag{10.9}$$

which will be useful shortly. Now expanding the field $\boldsymbol{U}$ in a Taylor series about $t + \phi\Delta t$ we obtain

$$\boldsymbol{U}(t+\Delta t) \approx \boldsymbol{U}(t+\phi\Delta t) + \frac{d\boldsymbol{U}}{dt}|_{t+\phi\Delta t}(1-\phi)\Delta t + \frac{1}{2}\frac{d^2\boldsymbol{U}}{dt^2}|_{t+\phi\Delta t}(1-\phi)^2(\Delta t)^2 + \mathcal{O}((\Delta t)^3) \tag{10.10}$$

and

$$\boldsymbol{U}(t) \approx \boldsymbol{U}(t+\phi\Delta t) - \frac{d\boldsymbol{U}}{dt}|_{t+\phi\Delta t}\phi\Delta t + \frac{1}{2}\frac{d^2\boldsymbol{U}}{dt^2}|_{t+\phi\Delta t}\phi^2(\Delta t)^2 + \mathcal{O}((\Delta t)^3). \tag{10.11}$$

Subtracting the two expressions yields

$$\frac{\boldsymbol{U}(t+\Delta t) - \boldsymbol{U}(t)}{\Delta t} \approx \boldsymbol{V}(t+\phi\Delta t) + \hat{\mathcal{O}}(\Delta t). \tag{10.12}$$

Inserting Equation 10.9 yields

$$\boldsymbol{U}(t+\Delta t) \approx \boldsymbol{U}(t) + (\phi\boldsymbol{V}(t+\Delta t) + (1-\phi)\boldsymbol{V}(t))\Delta t + \hat{\mathcal{O}}((\Delta t)^2). \tag{10.13}$$

and thus using Equation 10.8 yields

$$\boldsymbol{U}(t+\Delta t) \approx \boldsymbol{U}(t) + \boldsymbol{V}(t)\Delta t + \phi(\Delta t)^2\boldsymbol{L}(t+\phi\Delta t) + \hat{\mathcal{O}}((\Delta t)^2). \tag{10.14}$$

The term $\boldsymbol{L}(t+\phi\Delta t)$ can be handled in the following manner

$$\boldsymbol{L}(t+\phi\Delta t) \approx \phi\boldsymbol{L}(t+\Delta t) + (1-\phi)\boldsymbol{L}(t). \tag{10.15}$$

In summary, we have the following:

$$\boldsymbol{U}(t+\Delta t) \approx \boldsymbol{U}(t) + \boldsymbol{V}(t)\Delta t + \phi(\Delta t)^2(\phi\boldsymbol{L}(t+\Delta t) + (1-\phi)\boldsymbol{L}(t)) + \hat{\mathcal{O}}((\Delta t)^2). \tag{10.16}$$

We note that

- When $\phi = 1$, then this is the (implicit) Backward Euler scheme, which is very stable (very dissipative) and $\mathcal{O}((\Delta t)^2)$ locally in time,
- When $\phi = 0$, then this is the (explicit) Forward Euler scheme, which is conditionally stable and $\mathcal{O}((\Delta t)^2)$ locally in time,
- When $\phi = 0.5$, then this is the (implicit) "Midpoint" scheme, which is stable and $\hat{\mathcal{O}}((\Delta t)^2) = \mathcal{O}((\Delta t)^3)$ locally in time.

In summary, we have for the velocity[10]

$$\boldsymbol{V}(t+\Delta t) \approx \boldsymbol{V}(t) + \Delta t\,(\phi\boldsymbol{L}(t+\Delta t) + (1-\phi)\boldsymbol{L}(t)) \tag{10.17}$$

[10] In order to streamline the notation, we drop the cumbersome $\mathcal{O}(\Delta t)$-type terms.

and for the position

$$\begin{aligned}\boldsymbol{U}(t+\Delta t) &\approx \boldsymbol{U}(t) + \boldsymbol{V}(t+\phi\Delta t)\Delta t \\ &\approx \boldsymbol{U}(t) + (\phi\boldsymbol{V}(t+\Delta t) + (1-\phi)\boldsymbol{V}(t))\,\Delta t,\end{aligned} \tag{10.18}$$

or more explicitly

$$\begin{aligned}\boldsymbol{U}(t+\Delta t) \approx \boldsymbol{U}(t) + \boldsymbol{V}(t)\Delta t + \phi(\Delta t)^2 \\ (\phi\boldsymbol{L}(t+\Delta t) + (1-\phi)\boldsymbol{L}(t))\,.\end{aligned} \tag{10.19}$$

In iterative (recursion) form

$$\begin{aligned}\boldsymbol{U}^{L+1,K} = \underbrace{(\phi\Delta t)^2\boldsymbol{L}^{L+1,K-1}}_{\mathcal{G}(\boldsymbol{U}^{L+1,K-1})} \\ + \underbrace{\boldsymbol{U}^L + \boldsymbol{V}^L\Delta t + (\Delta t)^2\phi(1-\phi)\boldsymbol{L}^L}_{\mathcal{R}}\end{aligned} \tag{10.20}$$

Implicit time-stepping methods, with time-step size adaptivity, built on approaches found in Zohdi [41–43], will be used throughout the upcoming analysis.

References

1. Mueller M (2025) Dynamics and control of autonomous flight. Springer-Verlag
2. Tice BP (1991) Unmanned Aerial Vehicles – The Force Multiplier of the 1990s. Airpower Journal. When used, UAVs should generally perform missions characterized by the three Ds: dull, dirty, and dangerous
3. Alvarado Ed (2021) 237 Ways Drone Applications Revolutionize Business. Drone Industry Insights
4. Rekabi-Bana F, Hu J, Krajník T, Arvin F (2023) Unified Robust Path Planning and Optimal Trajectory Generation for Efficient 3D Area Coverage of Quadrotor UAVs IEEE Transactions on Intelligent Transportation Systems
5. Hu J, Niu H, Carrasco J, Lennox B, Arvin F (2023) Fault-tolerant cooperative navigation of networked UAV swarms for forest fire monitoring. Aerospace Science and Technology, (2022). Remote sensing of the environment using unmanned aerial systems (UAS). [S.l.]: ELSEVIER - HEALTH SCIENCE. ISBN 978-0-323-85283-8. OCLC 1329422815
6. Perks MT, Dal Sasso SF, Hauet A, Jamieson E, Le Coz J, Pearce S, Peña-Haro S, Pizarro A, Strelnikova D, Tauro F, Bomhof J, Grimaldi S, Goulet A, Hortobágyi B, Jodeau M (2020) Towards harmonisation of image velocimetry techniques for river surface velocity observations. Earth Syst Sci Data 12(3):1545–1559. https://doi.org/10.5194/essd-12-1545-2020.ISSN1866-3516
7. Koparan C, Koc AB, Privette CV, Sawyer CB (2020) Adaptive water sampling device for aerial robots. Drones 4(1):5. https://doi.org/10.3390/drones4010005
8. Koparan C, Koc A, Bulent; Privette, Charles V., Sawyer, Calvin B., Sharp, Julia L. (2018) Evaluation of a UAV-Assisted Autonomous Water Sampling. Water 10(5):655. https://doi.org/10.3390/w10050655
9. Koparan C, Koc A, Bulent; Privette, Charles V., Sawyer, Calvin B. (2018) In Situ Water Quality Measurements Using an Unmanned Aerial Vehicle (UAV) System. Water 10(3):264. https://doi.org/10.3390/w10030264
10. Koparan C, Koc A, Bulent; Privette, Charles V., Sawyer, Calvin B. (2019) Autonomous In Situ Measurements of Noncontaminant Water Quality Indicators and Sample Collection with a UAV. Water 11(3):604. https://doi.org/10.3390/w11030604
11. Cary L, Coyne J (2016) ICAO Unmanned Aircraft Systems (UAS), Circular 328. 2011–2012 UAS Yearbook – UAS: The Global Perspective (PDF). Blyenburgh & Co. pp. 112–115
12. Hu J, Lanzon A (2018) An innovative tri-rotor drone and associated distributed aerial drone swarm control". Robot Auton Syst 103:162–174. https://doi.org/10.1016/j.robot.2018.02.019
13. Garrow LA, German BJ, Leonard CE (2021) Urban air mobility: a comprehensive review and comparative analysis with autonomous and electric ground transportation for informing future research. Transp Res Part C Emerg Technol 132:103377. https://doi.org/10.1016/j.trc.2021.103377.ISSN0968-090X
14. Zhang C, Qiu Y, Chen J, Li Y, Liu Z, Liu Y, Zhang J, Hwa CS (2022) A comprehensive review of electrochemical hybrid power supply systems and intelligent energy managements for unmanned aerial vehicles in public services. Energy and AI 9:100175. https://doi.org/10.1016/j.egyai.2022.100175. hdl:10356/164036.ISSN2666-5468
15. Floreano D, Wood RJ (2015) Science, technology and the future of small autonomous drones. Nature 521(7553):460–466. https://doi.org/10.1038/nature14542
16. Fasano G, Accardo D, Tirri AE, Moccia A, De Lellis E (2015) Radar/electro-optical data fusion for non-cooperative UAS sense and avoid. Aerosp Sci Technol 46:436–450. https://doi.org/10.1016/j.ast.2015.08.010
17. Manfreda S, McCabe M, Miller P, Lucas R, Pajuelo MV, Mallinis G, Ben DE, Helman D, Estes L, Ciraolo G, Müllerová J, Tauro F, de Lima M, de Lima João, Maltese Antonino, (2018) On the Use of Unmanned Aerial Systems for Environmental Monitoring. Remote Sensing 10(4):641. https://doi.org/10.3390/rs10040641
18. Carlson DF, Rysgaard S (2018) Adapting open-source drone autopilots for real-time iceberg observations. MethodsX 5:1059–1072. https://doi.org/10.1016/j.mex.2018.09.003
19. Lesko J, Schreiner M, Megyesi D, Kovacs L (2019) Pixhawk PX-4 Autopilot in Control of a Small Unmanned Airplane. 2019 Modern Safety Technologies in Transportation (MOSATT). Kosice, Slovakia: IEEE. pp. 90–93. https://doi.org/10.1109/MOSATT48908.2019.8944101. ISBN 978-1-7281-5083-3. S2CID 209695691
20. Pierre-Jean B, François C, David V, Nicolas P (2011) The Navigation and Control technology inside the AR.Drone micro UAV. IFAC World Congress
21. Vinogradov E, Kumar AVSSB, Minucci F, Pollin S, Natalizio E (2023) Remote ID for separation provision and multi-agent navigation. 2023 IEEE/AIAA 42nd Digital Avionics Systems Conference (DASC). pp. 1–10. arXiv:2309.00843. https://doi.org/10.1109/DASC58513.2023.10311133
22. Balasingam M (2017) Drones in medicine-The rise of the machines. Int J Clin Pract 71(9):e12989. https://doi.org/10.1111/ijcp.12989.PMID28851081
23. Johnson AM, Cunningham CJ, Arnold E, Rosamond WD, Zègre-Hemsey JK (2021) Impact of using drones in emergency medicine: what does the future hold? Open Access Emerg Med 13:487–498. https://doi.org/10.2147/OAEM.S247020
24. Atherton KD (2023) The Marines are getting supersized drones for battlefield resupply. Popular Science. Keller, John (2023-12-13). Marine Corps orders 28 unmanned quadcopter aircraft for battlefield resupply in $11 million contract award. Military Aerospace
25. Dileep MR, Navaneeth AV, Ullagaddi S, Danti A (2020) A Study and Analysis on Various Types of Agricultural Drones and its Applications. 2020 Fifth International Conference on

Research in Computational Intelligence and Communication Networks (ICRCICN). IEEE. pp. 181–185. https://doi.org/10.1109/ICRCICN50933.2020.9296195.ISBN 978-1-7281-8818-8
26. Nazarov D, Nazarov A, Kulikova E (2023) Drones in agriculture: analysis of different countries. BIO Web of Conferences 67:02029. https://doi.org/10.1051/bioconf/20236702029. ISSN2117-4458
27. Liu J, Ding Y, Qiu R, Meng Z, Sun D, Peng X (2024) Aydin, Mucahit (ed.). Drone-Assisted Long-Distance Delivery of Medical Supplies with Recharging Stations in Rural Communities. Health & Social Care in the Community. 2024(1). https://doi.org/10.1155/2024/9143099. ISSN 0966-0410
28. Zohdi TI (2025) A machine-learning enabled digital-twin framework for tactical drone-swarm design. Comput Methods Appl Mech Eng. https://doi.org/10.1016/j.cma.2025.117999
29. Ring J (1963) The Laser in Astronomy. New Scientist, pp. 672-3
30. Cracknel, AP, Hayes L (2007) Introduction to Remote Sensing (2 ed.). London: Taylor and Francis. ISBN 0-8493-9255-1. OCLC 70765252
31. Goyer GG, Watson R (1963) The laser and its application to meteorology. Bull Am Meteor Soc 44(9):564–575
32. Medina A, Gaya F, Pozo F (2006) Compact laser radar and three-dimensional camera. J Opt Soc Am A 23:800–805
33. Trickey E, Church P, Cao X (2013) Characterization of the OPAL obscurant penetrating LiDAR in various degraded visual environments Proc. SPIE 8737, Degraded Visual Environments: Enhanced, Synthetic, and External Vision Solutions 2013, 87370E (16 May 2013); https://doi.org/10.1117/12.2015259
34. Hansard M, Lee S, Choi, O.& Horaud, R. (2012) Time-of-Flight Cameras: Principles. SpringerBriefs in Computer Science, Methods and Applications. https://doi.org/10.1007/978-1-4471-4658-2.ISBN978-1-4471-4657-5
35. Schuon S, Theobalt C, Davis J, Thrun S (2008) High-quality scanning using time-of-flight depth superresolution. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008. Institute of Electrical and Electronics Engineers. pp. 1–7
36. Gokturk SB, Yalcin H, Bamji C (2005) A Time-Of-Flight Depth Sensor - System Description, Issues and Solutions. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops. Institute of Electrical and Electronics Engineers: 35–45. https://doi.org/10.1109/CVPR.2004.291
37. ASC's 3D Flash LiDAR camera selected for OSIRIS-REx asteroid mission. NASASpaceFlight.com. 2012-05-13
38. Aue J, Langer D, Muller-Bessler B, Huhnke B, (2011–06-09). Efficient segmentation of 3D LiDAR point clouds handling partial occlusion. (2011) IEEE Intelligent Vehicles Symposium (IV). Baden-Baden Germany IEEE. https://doi.org/10.1109/ivs.2011.5940442.ISBN978-1-4577-0890-9
39. Hsu S, Acharya S, Rafii A, New R (2006) Performance of a Time-of-Flight Range Camera for Intelligent Vehicle Safety Applications. Advanced Microsystems for Automotive Applications 2006. VDI-Buch. Springer: 205–219. https://doi.org/10.1007/3-540-33410-6-16. ISBN 978-3-540-33410-1. Archived from the original (pdf) on 2006-12-06
40. Elkhalili O, Schrey OM, Ulfig W, Brockherde W, Hosticka BJ (2006) A 64x8 pixel 3-D CMOS time-of flight image sensor for car safety applications. European Solid State Circuits Conference 2006:568–571. https://doi.org/10.1109/ESSCIR.2006.307488,ISBN978-1-4244-0302-8
41. Zohdi TI (2002) An adaptive-recursive staggering strategy for simulating multifield coupled processes in microheterogeneous solids. Int J Numer Meth Eng 53:1511–1532
42. Zohdi TI (2010) Simulation of coupled microscale multiphysical-fields in particulate-doped dielectrics with staggered adaptive FDTD. Comput Methods Appl Mech Eng 199:79–101
43. Zohdi TI (2014) Embedded electromagnetically sensitive particle motion in functionalized fluids. Comput Part Mech 1:27–45
44. Zohdi TI (2018) A finite element primer for beginners. 2nd ed, Springer-Verlag
45. Zienkiewicz OC (1984) Coupled problems & their numerical solution, In: Lewis RW, Bettes P, Hinton E. (eds) Numerical methods in coupled systems Wiley, Chichester, pp. 35–58
46. Zienkiewicz OC, Paul DK, Chan AHC (1988) Unconditionally stable staggered solution procedure for soil-pore fluid interaction problems. Int J Numer Meth Eng 26:1039–1055
47. Lewis RW, Schrefler BA, Simoni L (1992) Coupling versus uncoupling in soil consolidation. Int J Num Anal Metho Geomech 15:533–548
48. Lewis RW, Schrefler BA (1998) The finite element method in the static and dynamic deformation and consolidation of porous media. 2nd ed. Wiley press
49. Schrefler BA (1985) A partitioned solution procedure for geothermal reservoir analysis. Comm Appl Num Meth 1:53–56
50. Turska E, Schrefler BA (1994) On consistency, stability and convergence of staggered solution procedures. Rend. Mat. acc. Lincei, Rome, S. 9, v.5, pp. 265-271
51. Bianco M, Bilardi G, Pesavento F, Pucci G, Schrefler BA (2003) A frontal solver tuned for fully coupled non-linear hygro-thermo-mechanical problems. Int J Numer Meth Eng 57:1801–1818
52. Wang X, Schrefler BA (1998) A multifrontal parallel algorithm for coupled thermo-hydro-mechanical analysis of deforming porous media. Int J Numer Meth Eng 43:1069–1083
53. Zohdi TI, Wriggers P (2008) Introduction to Computational Micromechanics. Springer-Verlag
54. Torquato S (2002) Random Heterogeneous Materials: Microstructure & Macroscopic Properties. Springer-Verlag, New York
55. Jikov VV, Kozlov SM, Olenik OA (1994) Homogenization of differential operators and integral functionals. Springer-Verlag
56. Hashin Z (1983) Analysis of composite materials: a survey. J Appl Mech 50:481–505
57. Mura T (1993) Micromechanics of defects in solids, 2nd edn. Kluwer Academic Publishers
58. Markov KZ (2000) Elementary micromechanics of heterogeneous media. In heterogeneous media: micromechanics modeling methods and simulations (Markov KZ, Preziozi L, Eds.), pp. 1 162. Birkhauser, Boston
59. Kachanov M (1993) Elastic solids with many cracks and related problems. Advance Applied Mechanics, vol. 30. Academic Press, New York, pp. 259
60. Kachanov M, Tsukrov I, Shafiro B (1994) Effective moduli of solids with cavities of various shapes. Appl Mech Rev 47:S151–S174
61. Kachanov M, Sevostianov I (2005) On the quantitative characterization of microstructures and effective properties. Int J Solids Struct 42:309–336
62. Sevostianov I, Gorbatikh L, Kachanov M (2001) Recovery of information of porous / microcracked materials from the effective elastic / conductive properties. Mater Sci Eng, A 318:1–14
63. Sevostianov I, Kachanov M (2008) Connections between Elastic and Conductive Properties of Heterogeneous Materials. Adv Appl Mech 42:69–253
64. Ghosh S (2011) Micromechanical analysis and multi-scale modeling using the voronoi cell finite element method. CRC Press/Taylor & Francis
65. Ghosh S, Dimiduk D (2011). Computational Methods for Microstructure-Property Relations. Springer NY
66. Zohdi TI. Computational optimization of the vortex manufacturing of advanced materials. Comput Methods Appl Mech Eng 190(46-47), 6231-6256

67. Zohdi TI, Monteiro PJM, Lamour V (2002) Extraction of elastic moduli from granular compacts. Int J Fract Letters Mech 115:L49–L54
68. Singh S, Singh R (2015) Study on Tribological Properties of Al–Al 2 O 3 Composites prepared through fdmaic route using reinforced sacrificial patterns. J Manuf Sci Eng 2015; 138(2):021009-021009-10. MANU-15-1023 https://doi.org/10.1115/1.4030772
69. Isaza C, Sierra G, Meza JM (2015) A novel technique for production of metal matrix composites reinforced with carbon nanotubes. J Manuf Sci Eng 138(2):024501. https://doi.org/10.1115/1.4030377
70. Khalifa NB, Foydl A, Pietzka D, Jager A (2015) Process limits of extrusion of multimaterial components. J Manuf Sci Eng 137(5):051001. https://doi.org/10.1115/1.4031091
71. Pacella M, Axinte DA, Butler-Smith PW, Shipway P, Daine M, Wort C (2015) An assessment of the wear characteristics of microcutting arrays produced from polycrystalline diamond and cubic boron nitride composites. J Manuf Sci Eng 138(2):021001. https://doi.org/10.1115/1.4030761
72. Bian D, Satoh G, Yao YL (2015) The laser interlaminar reinforcement of continuous glass fiber composites. J Manuf Sci Eng 137(6):061001. https://doi.org/10.1115/1.4030754
73. Decker BY, Gan YX (2015) Electric field-assisted additive manufacturing polyaniline based composites for thermoelectric energy conversion. J Manuf Sci Eng 137(2):024504. https://doi.org/10.1115/1.4029398
74. Dynamic thermomechanical modeling and simulation of the design of rapid free-form 3D printing processes with evolutionary machine learning Computer Methods in Applied Mechanics and Engineering 331:343-362
75. Gu D, Chang F, Dai D (2015) Selective laser melting additive manufacturing of novel aluminum based composites with multiple reinforcing phases. J Manuf Sci Eng 137(2):021010. https://doi.org/10.1115/1.4028925
76. Maxwell JC (1867) On the dynamical theory of gases. Philos Trans Soc London 157:49
77. Maxwell JC (1873) A treatise on electricity and magnetism, 3rd edn. Clarendon Press, Oxford
78. Rayleigh JW (1892) On the influence of obstacles arranged in rectangular order upon properties of a medium. Phil Mag 32:481–491
79. Hashin Z, Shtrikman S (1962) On some variational principles in anisotropic and nonhomogeneous elasticity. J Mech Phys Solids 10:335–342
80. Hashin Z, Shtrikman S (1963) A variational approach to the theory of the elastic behaviour of multiphase materials. J Mech Phys Solids 11:127–140
81. Powell D, Zohdi TI (2009) Attachment mode performance of network-modeled ballistic fabric shielding. Compos B Eng 40(6):451–460
82. Zohdi TI (2010) On the dynamics of charged electromagnetic particulate jets. Arch Comput Methods Eng 17(2):109–135
83. Zohdi TI (2013) Numerical simulation of charged particulate cluster-droplet impact on electrified surfaces. J Comput Phys 233:509–526
84. Zohdi TI (2012) Dynamics of charged particulate systems: modeling, theory and computation. Springer Science & Business Media
85. Duran J (1997) Sands, powders and grains. Springer Verlag, An introduction to the physics of Granular Matter
86. Pöschel T, Schwager T (2004) Computational granular dynamics. Springer Verlag
87. Onate E, Idelsohn SR, Celigueta MA, Rossi R (2008) Advances in the particle finite element method for the analysis of fluid-multibody interaction and bed erosion in free surface flows. Comput Methods Appl Mech Eng 197(19–20):1777–1800
88. Onate E, Celigueta MA, Idelsohn SR, Salazar F, Suárez B (2011) Possibilities of the particle finite element method for fluid-soil-structure interaction problems. Comput Mech 48:307–318
89. Rojek J, Labra C, Su O, Onate E (2012) Comparative study of different discrete element models and evaluation of equivalent micromechanical parameters. Int J Solids Struct 49:1497–1517. https://doi.org/10.1016/j.ijsolstr.2012.02.032
90. Carbonell JM, Onate E, Suarez B (2010) Modeling of ground excavation with the particle finite element method. J Eng Mech ASCE 136:455–463
91. Labra C, Onate E (2009) High-density sphere packing for discrete element method simulations. Commun Numer Methods Eng 25(7):837–849
92. Breder CM (1954) Equations descriptive of fish schools and other animal aggregations. Ecology 35(3):361–370
93. Beni G (1988) The concept of cellular robotic system. In IEEE International Symposium on Intelligent Control, pp. 57-62
94. Brooks RA (1991) Intelligence without reason. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI-91), pp. 569–595
95. Dudek G, Jenkin M, Milios E, Wilkes D (1996) A taxonomy for multi-agent robotics. Auton Robot 3:375–397
96. Cao YU, Fukunaga AS, Kahng A (1997) Cooperative mobile robotics: antecedents and directions. Auton Robot 4(1):7–27
97. Liu Y, Passino KM (2000) Swarm intelligence: literature overview. Technical report, Ohio State University
98. Turpin M, Michael N, Kumar V (2014) Capt: concurrent assignment and planning of trajectories for multiple robots, In: International Journal of Robotics Research, [https://journals.sagepub.com/doi/10.1177/0278364913515307]
99. Gazi V, Passino KM (2002) Stability analysis of swarms. Proceedings of the American Control Conference. Anchorage, AK May 8-10
100. Bender J, Fenton R (1970) On the flow capacity of automated highways. Trans Sci 4:52–63
101. Kennedy J, Eberhart R (2001) Swarm Intelligence. Morgan Kaufmann Publishers
102. Zohdi TI (2009) Mechanistic modeling of swarms. Comput Methods Appl Mech Eng 198(21–26):2039–2051
103. Zohdi TI (2018) Multiple UAVs for mapping: a review of basic modeling, simulation and applications. Annu Rev Environ Resour. https://doi.org/10.1146/annurev-environ-102017-025912
104. Zohdi TI (2019) The Game of Drones: rapid agent-based machine-learning models for multi-UAV path planning. Comput Mech 65(1):217–228
105. Goodrich P, Betancourt O, Arias AC, Zohdi TI (2023) Optimal placement of agricultural sensors using machine learning. Comput Electron Agric 205:107591
106. Bonabeau E, Dorigo M, Theraulaz G (1999) Swarm intelligence: from natural to artificial systems. Oxford University Press, New York
107. Dorigo M, Maniezzo V, Colorni A (1996) Ant system: optimization by a colony of cooperating agents. Syst Man Cyber Part B IEEE Trans 26(1):29–41
108. Bonabeau E, Meyer C (2001) Swarm intelligence: a whole new way to think about business. Harv Bus Rev 79(5):106–114
109. Fiorelli E, Leonard NE, Bhatta P, Paley D, Bachmayer R, Fratantoni DM (2004) Multi-auv control and adaptive sampling in monterey bay. In: Autonomous Underwater Vehicles, IEEE/OES, pp. 134–147
110. Feder T (2007) Statistical physics is for the birds. Physics Today p28-29
111. Ballerini M, Cabibbo N, Candelier R, Cavagna A, Cisbani E, Giardina I, Lecomte V, Orlandi A, Parisi G, Procaccini A, Viale M, Zdravkovic V (2008) Interaction ruling animal collective behavior

depends on topological rather than metric distance: evidence from a field study. PNAS 105(4):1232–1237
112. Zeyi Y (2024) Why china's dominance in commercial drones has become a global security matter, MIT Technology Review
113. Singh I (2024) The Secret to DJI's Drone Market Dominance: Revealed, DroneDJ
114. Chon G (2021) DJI Is a More Elusive U.S. Target Than Huawei, Reuters
115. Dress B (2024) China's Dominant Drone Industry Is a Step Ahead of Congress, The Hill
116. Whitepaper: AUVSI Partnership for Drone Competitiveness, AUVSI Partnership for Drone Competitiveness (2024)
117. Mozur P, Hopkins V (2024) Ukraine's War of Drones Runs Into an Obstacle: China, New York Times (2023; Hannah Beech & Paul Mozur, Drones Changed This Civil War, and Linked Rebels to the World, New York Times
118. Somerville Heather (2024) Why First Responders Don't Want the U.S. to Ban Chinese Drones, Wall Street Journal
119. Schönander L (2023) Securing the Skies: Chinese Drones and U.S. Cybersecurity Risks, Foundation for American Innovation
120. Timeline of U.S. Federal Government Activity Identifying and Addressing Unsecure UAS, Association for Uncrewed Vehicle Systems International (2024)
121. Holdeman E (2024) Federal government will require purchase of 'Made in America' drones, government technology
122. American Security Drone Act Of 2023, General Services Administration (last accessed 2024); Gallagher, Colleagues Introduce Bipartisan American Security Drone Act, The Select Committee on the Chinese Communist Party (2023)
123. Schneider J (2024) U.S. Department of the interior says anti-dji regulation hurt its operations, petapixel
124. Shepardson D (2025) US considers potential rules to restrict or bar chinese drones, Reuters
125. Agence France Presse, U.S. Drone Maker Says China Sanctions To Hit Supply Chain, Barron's (2024)
126. Somerville H (2024) American Drone Startup Notches Rare Victory in Ukraine, Wall Street Journal
127. Somerville H, Forrest B (2024) How American Drones Failed to Turn the Tide in Ukraine, Wall Street J
128. Wang J (2024) Taiwan Wants a Drone Army – but China Makes the Drones It Wants, Wall Street J
129. Buckley C, Chien AC (2024) Taiwan and U.S. Work to Counter China's Drone Dominance, New York Times
130. Alvarado Ed (2024) Commercial Use of Drone Swarms, Drone Industry Insights
131. Kallenborn Z (2024) Swarm Clouds on the Horizon? Exploring the Future of Drone Swarm Proliferation, Modern War Institute
132. Military and Security Developments Involving the People's Republic of China 2024, U.S. Department of Defense (2024)