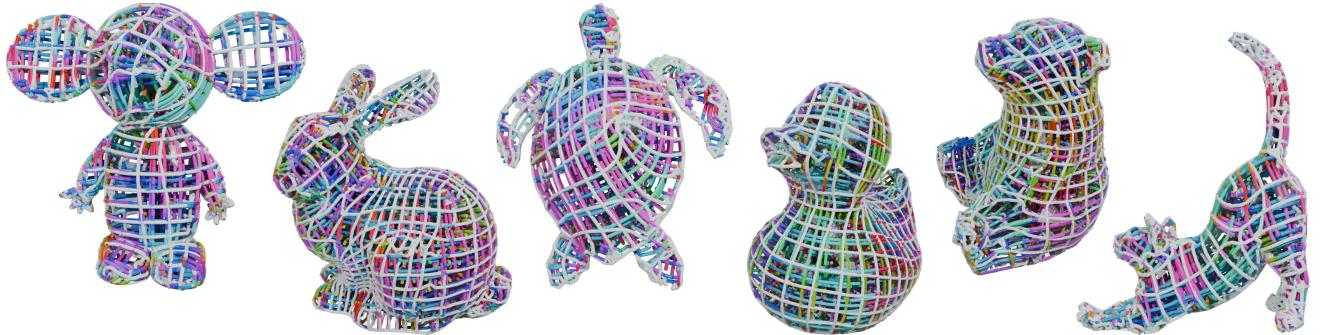


# Mint: Discretely Integrable Moments for Symmetric Frame Fields

J. Vekhter<sup>†1</sup>, Z. Chen<sup>2</sup> and E. Vouga<sup>‡1</sup>

<sup>1</sup>University of Texas at Austin, Austin, Texas, USA

<sup>2</sup>Adobe Inc., Seattle, WA, USA



**Figure 1:** We develop a novel approach to symmetric frame field design in volumes. Here we show integer grid parameterizations and frame field streamlines of approximately octahedral fields computed on a collection of animal models. Our method automatically recovers fields with complex topologies and is capable of producing parameterizations which conform to complex geometries in practice, as evidenced here.

## Abstract

This paper studies the problem of unconstrained (e.g. not orthogonal or unit) symmetric frame field design in volumes. Our principal contribution is a novel (and theoretically well-founded) local integrability condition for frame fields represented as a triplet of symmetric tensors of second, fourth, and sixth order. We also formulate a novel smoothness energy for this representation. To validate our discretization, we study the problem of seamless parameterization of volumetric objects. We compare against baseline approaches by formulating a smooth, integrable, and approximately octahedral frame objective in our discretization. Our method is the first to solve these problems with automatic placement of singularities while also enforcing a symmetric proxy for local integrability as a hard constraint, achieving significantly higher quality parameterizations, in expectation, relative to other frame field design based approaches.

**Keywords:** frame field optimization, integrability, volumetric parameterization

## 1 Introduction

Volumetric *symmetric frame fields*—generalizations of vector fields that assign to each point a smoothly-varying set of unlabeled vectors and their opposite vectors—arise naturally in applications

within and without computer graphics. In medicine, MRI machines collect volumetric data which can be post-processed into tensor fields and visualized as curves and surfaces for various clinical and research applications. For example, a frontier of medical imaging research are algorithms for diffusion MRI [LCD\*09], which produce tractography reconstructions of interwoven neural tissues. These reconstructions can be useful clinically for brain surgeons in planning operations that avoid severing vital neural pathways. In structural engineering, the finite element method can be used to predict the stress distribution of a part under load. This stress tensor field can be used as input into topology optimization pipelines to manufacture parts with improved strength-to-weight ratios [AJL\*19, KS23].

Symmetric frame fields have seen increasing use in the ge-

<sup>†</sup> This author would like to thank everyone who helped make this project possible, in particular Justin Solomon, Paul Zhang and David Palmer of the Geometric Data Processing Group at MIT who all significantly contributed to early stages of this work.

<sup>‡</sup> Supported by CMMI-2310666 and HCC-2212048

ometry processing community as *the* natural data structure for encoding orientation of a volumetric parameterization. In particular, many popular volume hexahedralization algorithms begin by optimizing for a symmetric frame field with “good” topological properties to use as a guide during the remeshing process [Eri14, LZC<sup>\*</sup>18, LB23]. A robust frame-field-based hexahedral meshing pipeline has remained out of reach, however, despite the considerable efforts of the community [PCS<sup>\*</sup>22] and significant interest from engineering fields.

**Key Challenge and Contribution** In the above applications, volumetric symmetric frame fields serve as an intermediate representation in a multi-step process of *frame field optimization* followed by *integration* of the frame field into isosurfaces or streamlines. A significant limitation of the optimize-then-integrate methodology is that, unlike in the vector field case where local integrability is a simple linear constraint that can be incorporated into the design optimization, in the *symmetric frame field* setting in volumes no analogous constraint has been formulated despite ongoing interest in this topic by the community [CCR24]. Consequently a designed symmetric frame field can be arbitrarily non-integrable and restoring integrability (such as by *trying* to integrate it using e.g. Cube-Cover [NRP11]) can require large, global adjustments to the field that violate design objectives.

Our work fills this gap: we show that discrete (local) integrability of a symmetric frame field on a tetrahedral volume mesh can be expressed as linear constraints on certain tensor representations of the field (where the entries of these tensors are polynomials in the natural representation of a frame field, as 9 DOFs per point).

## 1.1 Core Concepts and Definitions

As discussed above, in this paper we focus on symmetric frame fields:  $2^3$ -vector fields, in the nomenclature of Vaxman et al.’s [VCD<sup>\*</sup>16] taxonomy, on a 3D volume  $\Omega \subset \mathbb{R}^3$ . One straightforward representation of such a field is an assignment of a trio of vectors  $\{\mathbf{f}_1(\mathbf{p}), \mathbf{f}_2(\mathbf{p}), \mathbf{f}_3(\mathbf{p})\}$  to each point  $\mathbf{p}$  in  $\Omega$ , where it is understood that permuting the  $\mathbf{f}_i$  or flipping any of their signs yields the same symmetric frame. We will write  $\mathbf{F} \in \mathbb{R}^{3 \times 3}$  for the matrix with the  $\mathbf{f}_i$  as its columns.

The redundancy in this matrix representation can be understood in terms of a quotient space of  $\mathbb{R}^{3 \times 3}$ . Let  $\mathcal{O}$  be the octahedral group. Elements  $g \in \mathcal{O}$  have a representation in  $GL(3)$  as a product  $g = \mathbf{D}\mathbf{P}$  of a  $3 \times 3$  diagonal matrix  $\mathbf{D}$  with entries  $\pm 1$  and a permutation matrix  $\mathbf{P}$ . The octahedral group acts on matrices by right-multiplication,  $\mathbf{M} \mapsto \mathbf{MDP}$ , which permutes the columns of  $\mathbf{M}$  and alters their signs. Two matrices

$$\mathbf{F} = [\mathbf{f}_1 \quad \mathbf{f}_2 \quad \mathbf{f}_3], \quad \mathbf{G} = [\mathbf{g}_1 \quad \mathbf{g}_2 \quad \mathbf{g}_3]$$

thus represent the same  $2^3$ -vector frame if  $\mathbf{F} = \mathbf{Gg}$  for some  $g \in \mathcal{O}$ . In this case we write  $\mathbf{F} \sim \mathbf{G}$ . Following Palmer et al.’s [PBS20] similar notation for octahedral fields, we will write  $\mathbb{R}^{3 \times 3}/\mathcal{O}$  for the quotient space of 3-symmetric direction frames and  $GL(3)/\mathcal{O}$  for the space of non-degenerate (i.e., linearly independent) such frames. Note that  $GL(3)/\mathcal{O}$  is not a manifold, though it is locally

Euclidean in neighborhoods where all three directions  $\mathbf{f}_i$  have pairwise distinct magnitudes. A  $2^3$ -vector field is therefore a section  $\Gamma$  of  $\Omega \times \mathbb{R}^{3 \times 3}/\mathcal{O}$ .

Throughout the paper we will work with  $\mathbf{F}$  rather than its image under the quotient map, since the entries of  $\mathbf{F}$  are convenient degrees of freedom for numerical optimization. But of course a choice of vector trio  $\mathbf{F}$  is simply one representative out of several possible, and quantities such as smoothness, integrability, etc. must be measured with respect to  $\mathbb{R}^{3 \times 3}/\mathcal{O}$  rather than  $\mathbb{R}^{3 \times 3}$ .

In neighborhoods of  $\Omega$  away from  $\Gamma$ ’s *singular points and curves*, it is possible to *locally comb* the frame field, i.e. to find three smoothly varying vector fields  $\mathbf{c}_i$  with  $\mathbf{C} \sim \mathbf{F}$  at every point in the neighborhood. Just as with polyvector fields in two dimensions, it is generally not possible to extend this local combing to a global splitting of a symmetric frame field into three smooth vector fields due to the presence of topological dislocations along singular curves of the frame field.

**Discretization** In this paper we assume that  $\Omega$  is discretized as a tetrahedral mesh, and discretize symmetric frame fields as an assignment of a frame  $\mathbf{F}_i \in \mathbb{R}^{3 \times 3}/\mathcal{O}$  to every tetrahedron  $i$ .

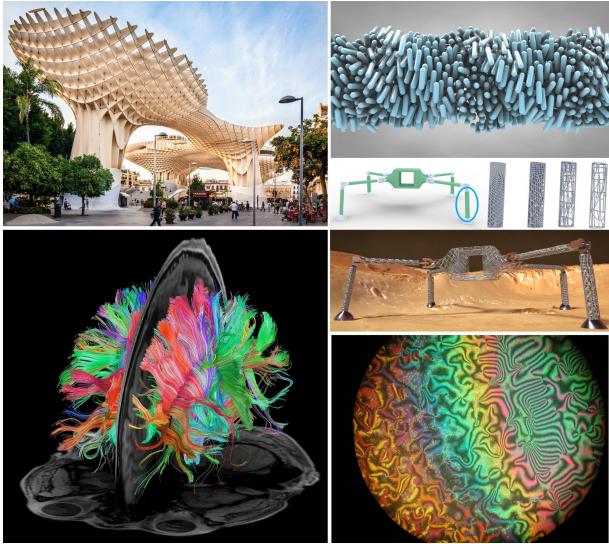
**Why  $GL(3)/\mathcal{O}$ ?** Most work in geometry processing, especially in the realm of 3D parameterization and hex meshing, focus on *octahedral* frames, with  $\mathbf{F}$  at every point a rotation and uniform scaling of the canonical Euclidean coordinate frame<sup>†</sup>. Forcing frames to be octahedral does have advantages in that it prevents appearance of highly-anisotropic frames, degenerate (coplanar or vanishing) frames, etc. Unfortunately, octahedral frames are *not* a good choice when it comes to integrability: an integrable octahedral frame implies existence of a local conformal parameterization. Under even mild additional constraints (such as alignment of frames with the boundary), the problem of finding an integrable octahedral frame becomes infeasible, due to the paucity of conformal maps in three dimensions (only Möbius transformations). For these reasons, in this paper we make the opposite design decision of enforcing integrability exactly and relaxing the octahedral property to a soft penalty, and evaluate the consequences of this decision on two related volumetric parameterization tasks.

## 2 Related Work

To our knowledge, this paper is the first to study the general problem of formulating a local integrability constraint for generic symmetric frame fields in volumes. However, we build upon a substantial body of literature covering related topics including representation, design, and integration of vector and frame fields.

Since our work is focused on volumetric frame fields, we do not comprehensively review papers on 2D vector fields, RoSys, or related structures. We refer to the surveys by de Goes et al. [dGLB<sup>\*</sup>14] and Vaxman et al. [VCD<sup>\*</sup>16] for more details.

<sup>†</sup> Note that octahedral *frames* should not be confused with the octahedral *group*; the latter describes the symmetries of all  $2^3$ -vector fields, of which the octahedral frame fields are a subset.



**Figure 2:** Some example applications of symmetric frame fields across scientific disciplines. Clockwise from top-left: the Metropol Parasol is one of the worlds largest wooden structures, in Seville, Spain (image courtesy of Flickr and Paul VanDerWerf). A rendering of the chiral nematic phase of liquid crystals (beautifulchemistry.net). An application of frame field optimization to the development of truss structures of varying densities, together with a render illustrating a potential application of these trusses for use in fabricating lightweight mars lander legs (Figure 17 of [AJL\*19]). An iridescent photo of the smectic (layered) liquid crystal phase which is modeled well by the Aviles-Giga PDE (Ji-Hoon Lee and the Case Western Liquid Crystal and Complex Fluids Group). Diffusion tensor imaging of the human brain [FTd\*14].

**Designing Volumetric Fields** Symmetric frame fields (i.e.  $2^k$ -vector fields) have been studied in both two [PPTSH14, DVPSH14] and three dimensions. The jump to three dimensions introduces substantial additional complexity, since the most convenient 2D representations exploit the geometry of the complex plane and do not have obvious 3D analogues. Moreover field singularities in 3D have much richer geometry (networks of singular curves and points) and topology than in 2D; several recent papers study this topology with the goal of identifying and, ideally, using local operations to repair singular structures that are unsuitable for applications like hexahedral meshing [LZC\*18, ZCFM23, LB23]. Algorithms for frame field design can be taxonomized based on whether they assume that the singular structure is provided, or try to optimize for the singularities automatically.

**Prescribed Singularities** Given the singular points of a symmetric frame field  $S$ , it can be *globally* combed on a covering space of  $\Omega \setminus S$  with topology given by the singular structure. This insight substantially simplifies the problem of designing frame fields in the case where the singular structure is prescribed in advance, either through use of manual tools [PRK\*17] or via a two-phase optimization [LLX\*12, LZC\*18]. Corman and Crane [CC19] use similar ideas to design octahedral frame fields given prescribed singular

structure. This latter work is primarily concerned with integrability of the octahedral frames in the sense of trivial monodromy rather than in the sense of volumetric parameterization, i.e. this work proposes an integrability constraint on the *derivative*, which is a necessary but not sufficient condition to guarantee field integrability. The major limitation of these approaches is the inability to optimize the singular structure jointly with the frame field in cases when a “good” singular structure isn’t known in advance.

**Automatic Methods** Many methods for symmetric frame field design with automatic singularity placement further restrict the frames to be octahedral, which allows leveraging representations of the frames based on the 4th band of spherical harmonics [HTWB11, RSL16, SVB17], 4th order tensors [CHRS19] or quaternions [GJTP17]. By enriching the 4th band of spherical harmonics with additional harmonics from the 2nd and 0th row, Palmer et al. [PBS20] allow for design of more general orthogonally decomposable (“odeco”) frames. An automatic method for designing *odeco* frame fields together with a symmetric integrability penalty was studied in 2D by Couplet et al. [CCR24], and preliminary work extending this formulation to 3D was documented in Couplet’s thesis [Cou24]. This formulation of symmetric integrability is implemented by expressing a vanishing Lie bracket condition in terms tensor coefficients, and developed *only* for *odeco* fields. Desobry et al.’s work [DCOC\*21] also uses direct products of the even bands of spherical harmonics to design unit frames that are not necessarily everywhere-orthogonal. In contrast, we believe we are first to propose a representation that can robustly handle arbitrary symmetric frame fields without constraints on norms or angles.

Metric-driven approaches [FHTB23, VS09] design frame fields in two passes, first computing a compatible metric  $g$  which encodes constraints like crease alignment and then solving for a smooth field relative to this metric. As integrability is not enforced during the frame field design step, the fields which result from this procedure will not be integrable. The one exception is when  $g$  is taken from a hex mesh already, but there is no proposed algorithm for solving for such a metric field in general in their paper. We evaluate against their implementation with  $g = I$ .

We stress that to our knowledge, *no* existing automatic method for frame field design guarantees locally integrable frame fields in any sense.

**Integrability Constraints** Local integrability constraints have been formulated for many discrete geometric objects other than symmetric frame fields, ranging from ordinary vector fields [PP03] to poly-vector fields [DVPSH15] to Chebyshev nets [SFCBCV19] to the conformal factor of conformal deformations [CPS15].

**Integration Algorithms** Given a designed symmetric frame field, the CubeCover algorithm [NRP11] uses mixed-integer programming to find a seamless or integer-grid parameterization of  $\Omega$  whose gradients match the frame field as closely as possible at each point. Related algorithms have been developed for similar problems, e.g. integration of 2D cross or frame fields [KNP07, BZK09] (possibly after rescaling to improve local integrability [RLL\*06, CIE\*16]), stripe patterns [KCPS15] or ribbons on surfaces [VZF\*19], vorticity fields [WPS14], etc. Single

vector fields in 3D can be projected to the nearest integrable field via Hodge decomposition [ZDWT19, RPPG19, CDG02]. While ex post facto projection and integration can sometimes yield good results in practice (particularly when integration error is small), the error in the resulting reconstruction is hard to predict or control compared to when integrability is included in the design process from the start.

**Applications** Symmetric frame fields (or more restrictive variants such as octahedral frame fields) are commonly used in field-guided approaches to volumetric parameterization and hexahedral meshing. Pietroni et al. [PCS<sup>\*</sup>22] survey various approaches, including methods for applying non-trivial local mesh repair operations [LBK16] necessary to realize robust hexahedral meshes in practice. The Aviles-Giga energy [AG87] constrains line fields to be everywhere unit and integrable and is a natural model of the smectic phase of liquid crystals. Researchers in that community [NWZ18] have developed their own computational methods for optimizing this energy and analyzing the resulting singular structures in symmetric fields [PPicv<sup>\*</sup>19] in parallel to the geometry processing community. Finally, Arora et al. [AJL<sup>\*</sup>19] propose designing truss structures that follow streamlines of volumetric stress, following an argument by Michell from 1901 that such a truss structure is optimally material-efficient. All of these applications benefit from the ability to design frame fields that are automatically locally integrable.

### 3 Overview

We discretize volumes  $\Omega$  as tetrahedral meshes and symmetric frame fields as frames  $\mathbf{F} \in \mathbb{R}^{3 \times 3}$  per tetrahedron. The entries of  $\mathbf{F}$  will also ultimately be our optimization variables in applications. However, as discussed in the Introduction, the matrices  $\mathbf{F}g$ , for  $g$  any element of  $\mathcal{O}$ , all correspond to the same symmetric frame in  $\mathbb{R}^{3 \times 3}/\mathcal{O}$ . An energy or constraint function  $f$  is therefore only well-defined if  $f(\mathbf{F}g) = f(\mathbf{F})$  for every element  $g$  of the octahedral group. To ensure that our optimization respects octahedral symmetry of the frames, we will define constraints and energies in terms of a (symmetric) three-tensor *moment representation* of the frame field  $L(\mathbf{F})$ , rather than in terms of the (not permutation invariant entries of)  $\mathbf{F}$  directly.

#### 3.1 Lifts and $k$ -th Moments

In regions where all three vector magnitudes are distinct, elements of  $GL(3)/\mathcal{O}$  have a unique representation as a matrix  $M_2 = \sum_i \mathbf{f}_i \mathbf{f}_i^T$  with the frame vectors as its eigenvectors with eigenvalues  $\|\mathbf{f}_i\|^2$ . On  $\mathbb{R}^{3 \times 3}/\mathcal{O}$  more generally this mapping is not injective, but the object  $M_2$  is nevertheless useful to work with since it can be easily computed from any representative frame  $\mathbf{F}$  and is invariant to the specific choice of representative frame (e.g. for any sign flips and permutation of the frame vectors  $\mathbf{FDP}$ , we have that  $M_2 = \sum_i \mathbf{f}_i \mathbf{DPP}^T \mathbf{D}^T \mathbf{f}_i^T$ , and the signs/permuations cancel out).

Tensor products of higher order can be used similarly to symmetrize  $\mathbf{F}$ . We write the *lift*  $L_k$  of a frame  $\mathbf{F}$  to  $k$ th order symmetric moment tensor as

$$L_k(\mathbf{F}) = \sum_i \mathbf{f}_i^{\otimes k}. \quad (1)$$

Note that when  $k$  is even, the lift satisfies the symmetry  $L_k(\mathbf{F}) = L_k(\mathbf{F}g)$  for all  $g \in \mathcal{O}$ ; the lift is thus a well-defined map from  $\mathbb{R}^{3 \times 3}/\mathcal{O}$  into  $\mathbb{R}^{3^k}$ . (And for this reason a variant of this lift operation has seen use [PBS20] for designing orthogonal frame fields.)

#### 3.2 Key Idea and Outline

We define an overall mapping from symmetric frames  $\mathbf{F}$  to the space of moments  $\mathcal{M} = \mathbb{R}^{3^2} \oplus \mathbb{R}^{3^4} \oplus \mathbb{R}^{3^6}$ :

$$L(\mathbf{F}) = [L_2(\mathbf{F}), L_4(\mathbf{F}), L_6(\mathbf{F})].$$

In Appendix A we prove that  $L$  is an injective map from  $\mathbb{R}^{3 \times 3}/\mathcal{O}$  to  $\mathcal{M}$ ; in other words, that  $\mathbf{F}$  can be uniquely recovered from  $\{L_k(\mathbf{F})\}_{k=2,4,6}$  up to octahedral symmetry. This property proves that  $\mathcal{M}$  is a suitable space on which to define optimization energies for symmetric frame fields, and will also be used to prove correctness of our local integrability constraint; the reader more interested in how to use our representation for optimization than in technical theory may skip this, but it is a key theoretical contribution of this work.

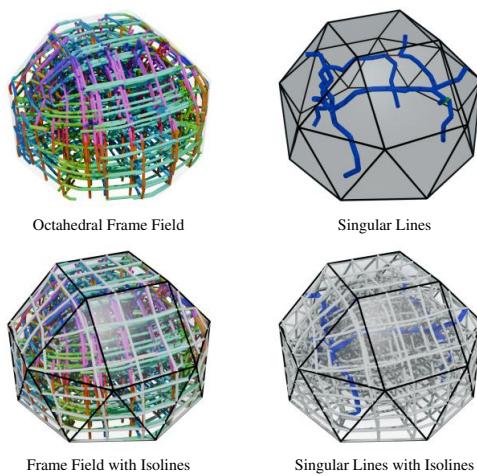
We will work with this moment representation of symmetric frames throughout the rest of the paper:

- We define local integrability of discrete symmetric frame fields in Section 4, as well as a notion of smoothness of symmetric frame fields.
- In Section 5, we introduce a model objective function (that we name MINT MESH) for recovering approximately octahedral fields for the application of computing boundary-aligned seamless and integer-grid parameterizations of 3D volumes that are as octahedral as possible.
- In Section 6, we document several nonstandard features of the numerical method we use to recover small residual minimizers of the MINT MESH optimization problem in practice.
- In Section 7 we distill a set of quality metric from the literature in order to evaluate the degree to which volume parameterizations are integrable, boundary aligned, and octahedral; and compare the fields produced by our method to those from several baseline algorithms.
- In Appendix B we give some motivation for studying the particular notion of smoothness over the symmetric frame representation that is introduced in this paper. In Appendix C we document some additional algorithmic details of our numerical method.

#### 4 Integrability of Symmetric Frame Fields

In this section we first define local integrability of symmetric frame fields in the smooth and discrete setting, then present our main contribution: a set of necessary and sufficient constraints on the representative vectors  $\mathbf{f}_i$  of a discrete symmetric frame field ensuring that the field is discretely locally integrable.

**Smooth Local Integrability** The notion of local combing allows us to immediately generalize local integrability of vector fields to  $2^3$ -vector fields: in particular, on a neighborhood  $\mathcal{N}$  where a symmetric frame field  $\mathbf{F}(\mathbf{p})$  can be combed—away from the field’s singularities, where there exists three smooth vector fields  $\mathbf{c}_i(\mathbf{p})$  with



**Figure 3:** This figure depicts the streamlines of a combed symmetric frame field on a polyhedral domain. The color map assigns colors to streamlines based on their orientation, meaning parallel flows share the same color—a technique inspired by DTI imaging [DHL09]. Top left: Streamlines of an approximately octahedral symmetric frame field on a polyhedral domain. Top right: The singular structure of the symmetric frame field. Bottom row: An overlay of the above visualizations with an integer grid parameterization of the symmetric frame field (see Section 7 for details on how we evaluate our solver). The key idea is to study a method for enforcing an integrability constraint on symmetric frame fields.

$\mathbf{C}(\mathbf{p}) \sim \mathbf{F}(\mathbf{p})$  for  $\mathbf{p} \in \mathcal{N}$ —the field is locally integrable if there exist three scalar potentials  $\phi_i : \mathcal{N} \rightarrow \mathbb{R}$  with  $\mathbf{c}_i = \nabla \phi_i$ ; or equivalently, if  $\nabla \times \mathbf{c}_i = 0$ .

**Discrete Local Integrability** Given a symmetric frame field on the tetrahedra of a volume mesh, we can ask, analogously to the smooth setting, for each vector field in a local combing of the frame field to be discretely integrable, i.e., the gradient of a discrete potential function  $\phi$  on the mesh’s vertices (extended to a piecewise-linear function on the volume via the hat basis). Following [PP03]’s characterization of discrete vector field integrability we can make this condition precise: a symmetric frame field is discretely locally integrable if, for every pair of tetrahedra sharing a common face  $\mathcal{F}$  there exists a  $\sigma \in \mathcal{O}$  with

$$\mathbf{f}_i \cdot \mathbf{e} = (\mathbf{G}\sigma)_i \cdot \mathbf{e} \quad (2)$$

for each  $i \in \{1, 2, 3\}$  and every edge  $\mathbf{e}$  on  $\mathcal{F}$ , where  $\mathbf{F}$  and  $\mathbf{G}$  are the representative vector frames on the two tets. In this definition  $\sigma$  plays the role of a local combing of the symmetric frame field on the two-tet “diamond” surrounding  $\mathcal{F}$ .

This definition can be used to formulate a meaningful measure of local symmetric integrability. On a tet mesh, we evaluate the min discrepancy in the above equation to obtain a scalar per facet:

$$Q_{\text{locint}} = \min_{\sigma \in \mathcal{O}} \frac{A(\mathcal{F})}{\ell(\mathcal{F})} \sum_{\mathbf{e} \in \mathcal{F}} \left( \sum_{i=1}^3 \|\mathbf{f}_i \cdot \mathbf{e} - (\mathbf{G}\sigma)_i \cdot \mathbf{e}\|^2 \right)^2 \quad (3)$$

with  $A(\mathcal{F})$  the area of the facet,  $\ell(\mathcal{F})$  the dual length between centroids.

The principle drawback of this characterization of discrete symmetric frame field integrability is that it cannot be easily enforced during frame field design, since Equation (2) involves a large number of discrete variables  $\sigma$  (one per interior face of the mesh). We next show how to reformulate these constraints into an equivalent form without this limitation. In Figure 10 we establish that our proposed formulation in Eq. (5) is an effective surrogate for Eq. (3).

#### 4.1 Moment-based Symmetric Integrability

In this section we will show how to express local integrability of a symmetric frame field in terms of  $L_k(\mathbf{F})$  and  $L_k(\mathbf{G})$  for frames  $\mathbf{F}$  and  $\mathbf{G}$  on two neighboring tets. The use of the moment tensors rather than the frames themselves will obviate the need for local combing, and allow us to express integrability as polynomial constraints on the entries of the representative frames  $\mathbf{F}_i$  with no additional discrete variables.

The key insight is the following: let  $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k$  be any set of vectors tangent to the common face  $\mathcal{F}$  of the two neighboring tetrahedra, and let  $\mathbf{M}_k = L_k(\mathbf{F})$  and  $\mathbf{N}_k = L_k(\mathbf{G})$ . Then if the field is discretely integrable, it must hold that for  $k$  even,

$$\begin{aligned} \mathbf{M}_k(\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k) &= \sum_i \prod_j \mathbf{f}_i \cdot \mathbf{v}_j \\ &= \sum_i \prod_j \mathbf{g}_i \cdot \mathbf{v}_j \\ &= \mathbf{N}_k(\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k). \end{aligned}$$

This fact follows from Equation (2) by expressing each vector  $\mathbf{v}_i$  in a basis of edges of  $\mathcal{F}$  and invoking invariance of  $\mathbf{N}_k$  to the choice of  $\sigma$ .

We can express the above equation compactly as  $\bar{\mathbf{M}}_k = \bar{\mathbf{N}}_k$  in terms of the restriction  $\bar{\mathbf{M}}_k = \mathbf{M}_k|_{\mathcal{F}}$  of  $\mathbf{M}_k$  to the plane tangent to  $\mathcal{F}$ . Equivalently, let  $\mathbf{\pi}_i$  be the projection of  $\mathbf{f}_i$  onto  $\mathcal{F}$ . If we choose an arbitrary basis for the tangent space of  $\mathcal{F}$  and write  $\bar{\mathbf{f}}_i \in \mathbb{R}^2$  for the expression of  $\mathbf{\pi}_i$  in this basis, we have as a necessary condition on discrete integrability that for all even  $k$ ,

$$L_k(\bar{\mathbf{F}}) = L_k(\bar{\mathbf{G}}), \quad (4)$$

where  $\bar{\mathbf{F}} \in \mathbb{R}^{2 \times 3}$  is the matrix with columns  $\bar{\mathbf{f}}_i$ .

In practice we choose to enforce this constraint via a penalty on its residual when  $k = 2, 4$ , and  $6$ :

$$E_{\text{int}} = \frac{A(\mathcal{F})}{\ell(\mathcal{F})} \sum_{\mathcal{F}} \sum_{k=1}^3 \|L_{2k}(\bar{\mathbf{F}}) - L_{2k}(\bar{\mathbf{G}})\|_F^2, \quad (5)$$

where the outer sum is over all interior facets  $\mathcal{F}$  of the tet mesh;  $\mathbf{F}$  and  $\mathbf{G}$  are the frames on the tets neighboring  $\mathcal{F}$ ,  $A(\mathcal{F})$  is the area of the facet,  $\ell(\mathcal{F})$  is the dual length between centroids and  $\|\cdot\|_F$  is the Frobenius norm.

**Sufficiency** Intuitively, one might expect that enforcing Equation (4) for sufficiently many different values of  $k$  constrains  $\mathbf{F}$  and  $\mathbf{G}$  to be locally integrable in the sense of Equation (2). In fact, this is true, and it is enough to enforce Equation (4) for  $k \in \{2, 4, 6\}$  as in Equation (5). For if  $L(\bar{\mathbf{F}}) = L(\bar{\mathbf{G}})$  then it follows from injectivity of  $L$  (and suitable zero-padding of the tensors in question) that  $\bar{\mathbf{F}} \sim \bar{\mathbf{G}}$  and therefore Equation (2) holds. The counterexample in Equation (25) (where without loss of generality we can take  $\mathcal{F}$  to be parallel to the  $xy$  plane) shows that enforcing Equation (4) for  $k = 2, 4$  is not enough to guarantee local integrability of the discrete frame field.

**Implementation Notes** Naively, each tensor in Equation (4) has  $2^{2k}$  entries; however many of these are redundant due to the tensor symmetries. Grouping together redundant constraints leaves

$$\binom{2}{2} + \binom{2}{4} + \binom{2}{6} = 15$$

total unique tensor coefficients per interior face of the tet mesh that must be computed to evaluate Equation (5).

## 4.2 Compatible Smoothness Operator

Our method is called MINT for the Moment based INTegrability constraint discretized in section 4.1. However, this constraint is not enough on its own to formulate interesting symmetric frame field design objectives. One basic, but not entirely straightforward, question is how to “correctly” formulate a notion of smoothness over  $\mathbb{R}^{3 \times 3}/\mathcal{O}$ . In this paper studied a Laplacian acting on the same injective representation of  $\mathbb{R}^{3 \times 3}/\mathcal{O}$  we use to enforce integrability:

**Higher Order Moment Laplacian** Following Palmer et al. [PBS20], we regularize smoothness of symmetric frame fields using a Dirichlet-like energy  $\|\nabla L_k(\mathbf{F})\|_F^2$  on the moment tensors. Accordingly we propose the following discrete smoothness energy, written using the notation of Section 4:

$$E_{\text{smooth}} = \sum_{\mathcal{F}} \frac{A(\mathcal{F})}{2\ell(\mathcal{F})} \sum_{k=1}^3 \|L_{2k}(\mathbf{F}) - L_{2k}(\mathbf{G})\|_F^2, \quad (6)$$

with  $A(\mathcal{F})$  the area of facet  $\mathcal{F}$ ,  $\ell(\mathcal{F})$  is the dual length between centroids

See Appendix B for additional discussion of this Laplacian.

## 5 MINT MESH: Formulation

Much previous work on symmetric frame field design is motivated by the task of locally-injective seamless volume parameterization: an extension of  $uv$ -mapping to volumes  $\Omega \subset \mathbb{R}^3$  with applications in hexahedral meshing, medical imaging, and computational fabrication. In this section, we present an optimization algorithm for designing a symmetric frame field  $\mathbf{F}$  to guide seamless parameterization. This algorithm exploits our moment-based frame field representation to promote local integrability of  $\mathbf{F}$  during the design process. We then globally integrate the designed  $\mathbf{F}$  into a parameterization  $\phi$  using the CubeCover algorithm [NRP11].

We follow previous work [LB23] in considering two variations of the parameterization task. *Seamless parameterization* requires that the image of each singular line of  $\phi$  has two integer coordinates (with singular points at the intersection of singular lines mapping to points of the lattice  $\mathbb{Z}^3$ ). An *integer-grid* parameterization additionally requires that the image of every point of  $\partial\Omega$  has an integer coordinate; this latter constraint is necessary to construct from the parameterization a hexahedral mesh.

Below, we formulate several additional optimization terms useful when trying to recover approximately octahedral parameterizations. We then combine these terms, together with our notions of integrability and smoothness to specify a concrete optimization problem for computing boundary-aligned seamless and integer-grid parameterizations of 3D volumes that are as octahedral as possible (MINT MESH).

All of these energies are ultimately implemented as functions of the discrete frame field  $\mathfrak{F}$ , represented as per-tetrahedron frames  $\mathbf{F}$  with frame vectors  $\mathbf{f}_i$ . By using energies obviously invariant under the action of  $\mathcal{O}$  on the frames, we ensure the energies are well-defined irrespective of the choice of representative frame  $\mathbf{F}$  on each tetrahedron.

## 5.1 Approximately Octahedral Fairing

Here we develop three additional objective terms we use to regularize our frame field optimization towards approximately octahedral solutions. We note that there is still considerable research remaining in designing interesting design objectives for symmetric frame field design. One particularly natural extension we don’t consider is a log barrier term to prevent inverted elements, as often used in the parametrization literature.

**Orthogonality** The energy

$$E_{\text{orth}} = \sum_{\mathcal{T}} \frac{V(\mathcal{T})}{2} \sum_{i \neq j} (\mathbf{f}_i \cdot \mathbf{f}_j)^2 \quad (7)$$

promotes symmetric frame fields whose vectors are pairwise orthogonal (“odeco”), where the sum is over all tetrahedra  $\mathcal{T}$  with volume  $V(\mathcal{T})$  and associated frame  $\mathbf{F}$ .

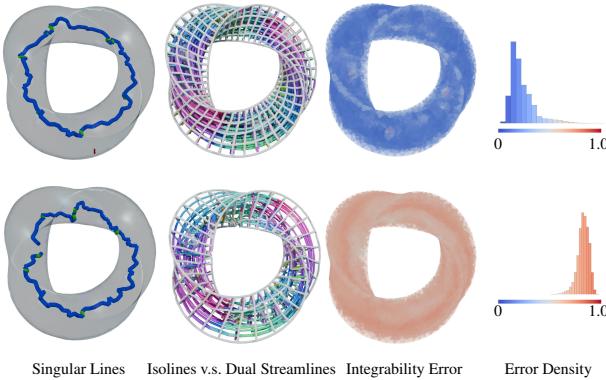
**Unit Norm** As a fairness term to promote symmetric frame fields whose vectors are as-unit-as-possible, we propose the straightforward

$$E_{\text{unit}} = \sum_{\mathcal{T}} \frac{V(\mathcal{T})}{2} \sum_{i=1}^3 (\|\mathbf{f}_i\|^2 - 1)^2. \quad (8)$$

**Plane Alignment** Given a prescribed direction  $\mathbf{d}$ , the following energy penalizes deviation of a frame vector  $\mathbf{f}_i$  from the plane orthogonal to  $\mathbf{d}$ . In other words, this energy requires the isosurface of one scalar potential  $\phi$  to contain  $\mathbf{d}$ :

$$E_{\text{plane}}(\mathcal{T}, \mathbf{d}, i) = \frac{V(\mathcal{T})}{2} (\mathbf{d} \cdot \mathbf{f}_i)^2, \quad (9)$$

For  $E_{\text{plane}}$  we place one energy term for each boundary facet which penalizes two of the frame vectors from leaving the tangent



**Figure 4:** The integer grid parametrization on a twisted triangle torus model. In the top row, frame fields are computed using MINT MESH. In the bottom they are computed using the same energy model without an integrability constraint, denoted as MINT OCTA. Here we see a clear qualitative improvement in integer grid parametrization by applying our symmetric integrability formulation.

plane. This is the same boundary condition as is used in the baseline octahedral frame field solver from [FHTB23].

## 5.2 An As-Octahedral-As-Possible Objective

Putting the above terms together allows us to formulate the following fairness objective:

$$E_{\text{mesh}} = E_{\text{smooth}} + \lambda_{\text{orth}} E_{\text{orth}} + \lambda_{\text{unit}} E_{\text{unit}} + \lambda_{\text{plane}} E_{\text{plane}}. \quad (10)$$

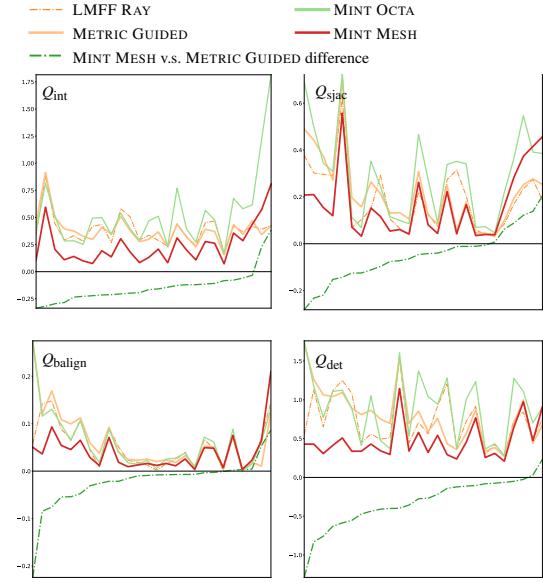
In this paper, for the sake of doing a concrete experiment, we study this objective with  $\lambda_{\text{orthog}} = 0.1$ ,  $\lambda_{\text{unit}} = 10^{-5}$ , and  $\lambda_{\text{plane}} = 10^{-6}$ .

We enforce an additional hard constraint that one of the frame vectors is aligned to the surface normal at the volume boundary. To avoid over-constraining the problem at sharp edges and corners, we add a *ghost tetrahedron* adjacent to every boundary facet of the input mesh, each with its own frame, and we associate the normal  $\hat{\mathbf{n}}(\mathcal{T})$  of the boundary facet to the ghost tetrahedron  $\mathcal{T}$ . In Appendix C, we summarize additional subdivision rules which we apply to pre-process an input mesh to ensure that no tet is adjacent to more than one ghost tet.

The full MINT MESH optimization problem is then:

$$\begin{aligned} \arg \min_{\mathfrak{F}} E_{\text{mesh}} \quad & \text{s.t.} \\ E_{\text{int}} &= 0, \\ \mathbf{f}_1(\mathcal{T}) &= \hat{\mathbf{n}}(\mathcal{T}) \quad \forall \text{ghost tets } \mathcal{T}. \end{aligned} \quad (11)$$

For our experiments in Section 7, we also consider a variant of Equation (11) without the integrability constraint, which we term MINT OCTA. Both MINT OCTA and MINT MESH depart from previous work in only enforcing that frames be “as octahedral as possible,” allowing anisotropy and non-orthogonality in the solutions as necessary. We will establish that MINT OCTA is comparable with



**Figure 5:** We compare the performance of our proposed method by evaluating the quality of integer grid parameterizations obtained on models from the HexMe1 dataset of industrial models. To ensure consistency, we remesh these models using TetWild, targeting a lower resolution of approximately 3k tetrahedra. Notably, the two baseline solvers do not produce identical results on these meshes, despite both implementing a tetrahedral-based version of [RSL16]. These plots assess performance across various metrics discussed in Section 7.1. On this practical dataset, our method produces boundary-aligned parameterizations that significantly outperform the baselines with respect to these quality metrics.

baseline frame field design methods for the seamless and integer-grid parameterization tasks, and that MINT MESH in many cases improves the parameterization quality.

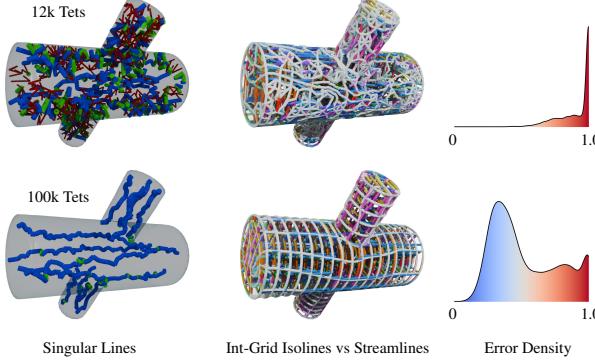
In Section 6, we provide more details on the numerical algorithm which we use to recover solutions to this problem in practice. In Figure 4 we illustrate how our symmetric integrability constraint changes the result of integer grid parameterization on a twisted triangular torus model.

## 6 MINT MESH: Numerical Method

On a high level, our approach is to apply Newton’s method to directly solve the (highly) non-linear optimization problem in Equation (11). In this section, we expand on some essential algorithmic details of our numerical method. More implementation details are available in Appendix C, and source code for a reference implementation of our method is available at <https://github.com/the13fools/Mint3D>.

### 6.1 Problem Setup

While we needed to make a concrete choice of parameters to perform our experiments, results on specific models may benefit



**Figure 6:** In this figure we probe the sensitivity of MINT MESH to mesh resolution, on the *s08o\_cross\_cyls\_dr* model from the HexMes dataset. Our default formulation produces a result with many degenerate frames on the relatively low resolution mesh included with the dataset, but improves dramatically after using TetWild [HZG\*18] to remesh the geometry to one with approximately 10x the number of tets. We plot the results relative to integer grid parameterization. (Top), we show the results on a mesh with 12,638 tetrahedra, driven to full convergence (86 newton iterations). (Bottom), we perform the solve on a mesh with approximately 10 times more elements (e.g. 101061 tetrahedra), showing partially converged results (30 newton iterations - where the weight on the  $E_{int}$  term is approx 5x the weight on  $E_{smooth}$ ).

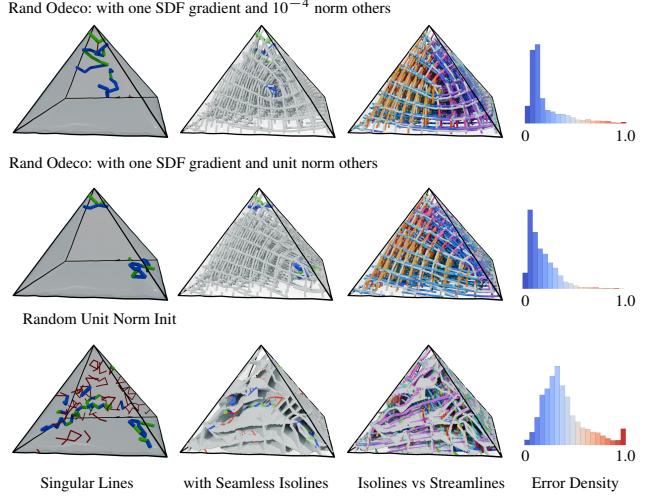
from increasing the mesh resolution, adjusting hyperparameters, and changing the initialization.

**Resolution Dependence** While we made an effort to include scaling coefficients in our energies so that they converge under mesh refinement, our default choice of parameters may produce frame fields with degeneracies on low resolution/low quality input meshes. In Figure 6, we illustrate how frame field quality from MINT MESH improves under subdivision. This example was used as a challenging test case in Couplet’s thesis [Cou24] (as documented in Fig. 4.12 of that work). Figure 19 documents how fields depend on model parameters on this geometry.

**Choice of Hyperparameters** The quality of the MINT MESH results can be improved by tuning the weights of the various terms Equation (10) (as we illustrate in Figure 19). We include additional discussion on how to tune these weights in Appendix C. In particular, the presence/absence of degenerate (zero-volume) and inverted (negative-volume) frames is sensitive to these parameter choices, suggesting that the robustness of our formulation could be improved by including a barrier penalty on volume inversion in the objective function.

**Sensitivity to Initialization** The problem we are solving is non-convex, and the minimum that our reference implementation converges to is *initialization dependent*. We illustrate an example of this initialization dependence in Figure 7.

For evaluation in Section 7, we chose an initialization strategy that we found yields relatively reliable results. We first compute a



**Figure 7:** In this figure we show the impact of different initializations on the converged results of MINT MESH. In the top row we illustrate the result from our default configuration, where one frame is initialized to a signed distance field, and the other two are initialized to be random orthogonal vectors with norm  $10^{-4}$ . In the center row, we initialize in the same way, except the vectors orthogonal to the SDF gradient are set to have unit norm. In the bottom row, we initialize all frame vectors to be random with unit norm. We plot the results relative to seamless integration in this figure (e.g. without integer boundary constraints). This experiment demonstrates that on some examples our formulation may converge to qualitatively different minima depending on initialization.

signed distance field (SDF)  $d$  on the interior of each mesh, and set  $\mathbf{f}_1 = \nabla d$  on each interior (non-ghost) tetrahedron. We initialize the other two frame vectors to two random vectors with norm  $10^{-4}$  that are orthogonal to each other and to the SDF gradient. Recall that on ghost tets, we set  $\mathbf{f}_1$  to the normal of the adjacent mesh boundary facet as a hard constraint. For ghost tets adjacent to a sharp surface feature, we initialize one of the two other frame vectors to be in the tangent plane perpendicular to the feature edge with norm 25.

We find that the energy landscape of the  $E_{smooth}$  objective is often locally convex despite the high order of the expression (see Figure 18)—minimizing smoothness on its own will converge reliably to the same solution from a wide basin of initial conditions. As such, choosing a fairness term with a high smoothness weight generally reduces the initialization dependence of solutions. For our choice of hyperparameters, Equation (11) converges to similar qualitative solutions for a wide range of initializations on many models, but in Figure 7 we illustrate an example where this is not the case.

We also tried setting the frame vectors orthogonal to the SDF gradient to have unit norm at initialization rather than norm  $10^{-4}$ . While this choice typically leads to the solver to converge in significantly fewer iterations, and in some cases to solutions with lower energy, fields resulting from this choice of initialization are more

likely to converge to qualitatively different minima in repeated trials, and so we did not use it in our experiments.

## 6.2 Integrability Constraint Enforcement

Recovering solutions to Eq. 11 with small constraint residual is not straightforward.

The tensors in Equation (4) are polynomials of up to degree 6 in the frame vector coefficients, and standard optimization packages have limited support for general polynomial constraints in large, sparse problems. An additional numerical challenge is that the constraint has a large null space. This is a property shared by per-simplex discretizations of vector field integrability as well [PP03, TLHD03]. Applying a black box numerical linear algebra method (e.g. LUQ decomposition [Jac]) to remove the constraint kernel made the constraint matrix significantly denser and degraded overall performance in our experiments, but remains an intriguing idea for future work.

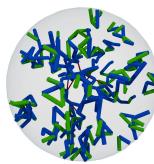
**Deflated Penalty Method** We instead take the approach of enforcing integrability constraints via a sequence of sub-problems, where each sub-problem solves Equation (11) using a soft integrability penalty potential with incrementally increasing relative stiffness. Specifically, at iteration  $i$  of the algorithm, we solve for  $\mathfrak{F}^{i+1}$  by:

$$\begin{aligned}\mathfrak{F}^{i+1} &= \arg \min_{\mathfrak{F}} 10^8 \left( \frac{1}{\lambda_{\text{pen}}} \right)^i E_{\text{mesh}} + E_{\text{int}} \quad \text{s.t.} \\ f_1(\mathcal{T}) &= \hat{\mathbf{n}}(\mathcal{T}) \quad \forall \text{ ghost tets } \mathcal{T}.\end{aligned}\quad (12)$$

using the previous solution  $\mathfrak{F}^i$  as the initial guess (or the initialization described in the previous section for  $\mathfrak{F}^0$ ). The optimization problem at each iteration is solved using Newton’s method. We set  $\lambda_{\text{pen}} = \sqrt{2}$  and terminate the procedure when  $(\lambda_{\text{pen}})^i > 10^{25}$ . (Note that scaling down  $E_{\text{mesh}}$  is mathematically equivalent to scaling up  $E_{\text{int}}$ ; we choose our formulation to maintain numerical stability of the Newton solve as  $i$  increases.)

For this approach, due to the high-order nature of  $E_{\text{int}}$ , an *extremely* small residual is required to ensure that it translates to a useful bound on local integrability in the sense of Equation (2). In Figure 10, we plot the average constraint residual and local integrability residual over our data sets with and without constraint enforcement. This figure establishes degree to which this numerical method succeeds in reducing the constraint residual relative to an ablation, as well as the extent to which this reduction in constraint residual translates to a reduction in local integrability residual.

**Regularizing the Constraint** Directly solving Equation (12) surfaces a numerical instability. When the penalty term gets large the optimization problem begins to become underdetermined. In the limit where there is no  $E_{\text{mesh}}$  term, all integrable frame fields have equal energy. In practice we begin to observe discontinuities in between iterations as the solver “jumps” from one integrable frame field to another around when  $(\lambda_{\text{pen}})^i > 10^{12}$ , i.e. the factor in front of  $E_{\text{mesh}}$  is smaller than .001). We illustrate what the singular structure of



such a field looks like at this point without any regularization on the inset sphere.

A simple and effective approach to addressing this issue is to place a large diagonal regularizer on the constraint. We replace  $E_{\text{int}}$  in Equation (12) with:

$$\tilde{E}_{\text{int}} = E_{\text{int}} + \lambda_{\text{reg}} I. \quad (13)$$

In our experiments, we set  $\lambda_{\text{reg}} = 10^{-1}$ ; intuitively, this diagonal regularizer adds “viscosity” to the optimization procedure and prevents the solution from jumping to spurious integrable frame fields once the  $E_{\text{mesh}}$  coefficient vanishes. One limitation of this approach is that it slows down convergence. There is room to improve this formulation with additional research (an adaptive weighting strategy, sometimes used for physical simulation of dynamics (e.g. Andrews et al. [ATK17]), might be a promising direction to explore).

## 7 Quantitative Evaluation

In this section, we design an experiment to empirically evaluate the performance of MINT MESH relative to baseline methods for designing frame fields in volumes for seamless parameterization.

In Section 7.1, we distill several parameterization quality metrics from the literature. In Section 7.2 we detail the datasets that we evaluate on. In Section 7.3 we summarize the baseline methods for octahedral frame field design that we evaluate against in our study. In Section 7.4 we point out several observations from our quantitative evaluation.

### 7.1 Parameterization Quality Metrics

**Global Integrability Error** We measure how large of a change in the designed frame field was required to globally integrate it:

$$Q_{\text{int}} = \min_{g \in \mathcal{O}} \frac{1}{3} \sum_{i=1}^3 \|\nabla \phi_i - (\mathbf{F}g)_i\|^2, \quad (14)$$

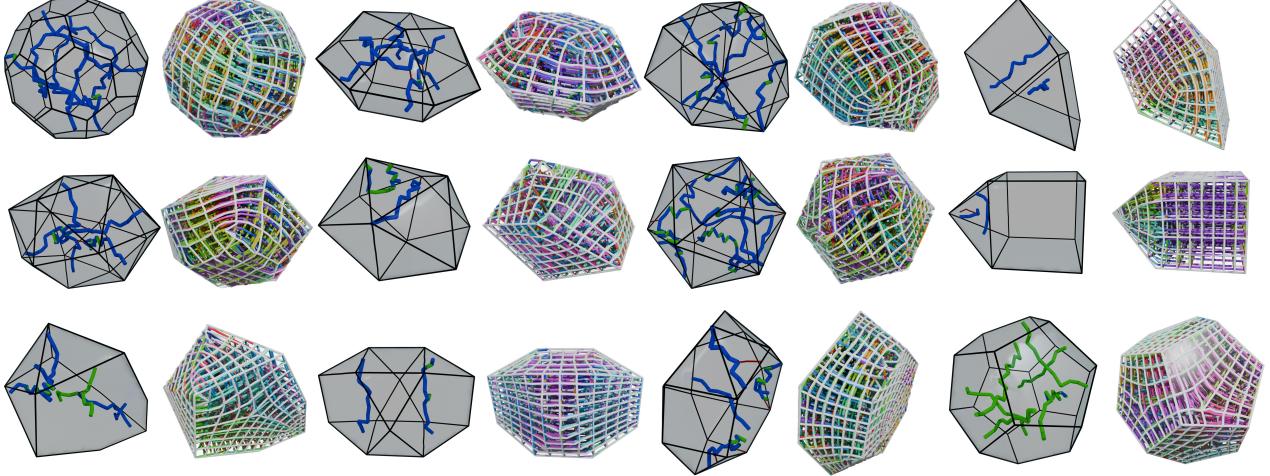
where  $\phi_i$  are the integrated coordinate functions of the parameterization. While the global integrability error does not directly measure the quality of the parameterization, a large integrability error generally correlates with CubeCover inserting a large amount of distortion or a topology change in the parameterization.

As we will see below, enforcing the condition from Eq. 4 as a constraint consistently translates to a measurable decrease in  $Q_{\text{int}}$ .

Note that this is the only metric that depends on the designed frame field  $\mathbf{F}$ ; all subsequent metrics are only properties of the final volume parameterization  $\phi$ .

**Scaled Jacobian** The hexahedral meshing research community has extensively studied how to evaluate parameterizations intended for meshing applications [GHX<sup>\*</sup>17]. One common [LB23] metric is the scaled Jacobian, which measures “skew” of the parameterization, i.e. the failure of the parameterization to have orthogonal isosurfaces:

$$Q_{\text{jac}} = \frac{1}{|\mathcal{T}|} \sum_{\mathcal{T}} \left( 1 - \frac{|\det(J\phi)|}{\prod_{i=1}^3 \|\nabla \phi_i\|} \right). \quad (15)$$



**Figure 8:** A gallery showcasing parametrizations and dual field streamlines generated by MINT MESH on our polyhedral dataset. For each model, the left column displays the singular structures, while the right column presents the integer grid parametrization (isolines) overlaid with the dual streamlines of the optimized frame fields.

**Anisotropy** Complementary to the scaled Jacobian, anisotropy measures the relative difference between the largest and smallest coordinate function gradient at each point:

$$Q_{\text{aniso}} = \frac{1}{|\mathcal{T}|} \sum_{\mathcal{T}} \frac{M - m}{M}, \quad (16)$$

$$M = \max_{i=1}^3 \|\nabla \phi_i\|, \quad m = \min_{i=1}^3 \|\nabla \phi_i\| \quad (17)$$

Together these two metrics capture angle distortion of the parameterization:  $Q_{\text{jac}} = Q_{\text{aniso}} = 0$  if and only if  $\phi$  is conformal.

**Determinant Deviation** We compare the determinant of the parameterization Jacobian on each tetrahedron to the determinant averaged over the entire mesh to quantify the global volume distortion of  $\phi$ :

$$Q_{\text{det}} = \frac{1}{|\mathcal{T}|} \sum_{\mathcal{T}} \left| \frac{|\det(J\phi)| - A}{A} \right|, \quad (18)$$

$$A = \frac{1}{|\mathcal{T}|} \sum_{\mathcal{T}} |\det(J\phi)|. \quad (19)$$

Note that, like all of our quality metrics,  $Q_{\text{det}}$  is invariant to constant global scaling of the parameterization.

**Fraction of Inverted Elements** We measure failure of a parameterization to be locally injective by calculating the fraction of tetrahedra with  $\det(J\phi) < 0$  (after globally reflecting the parameterization so that the majority of elements have positive Jacobian determinant).

**Combed Smoothness** We evaluate the smoothness of the parameterization by computing the Dirichlet energy of the parameterization gradient:

$$Q_{\text{smooth}} = \sum_{\mathcal{F}} \frac{A(\mathcal{F})}{2} \sum_{i=1}^3 \left\| \nabla \phi_i^1 - \nabla \phi_i^2 \right\|^2, \quad (20)$$

where the sum is over interior facets  $\mathcal{F}$  with area  $A(\mathcal{F})$ , and  $\phi^1$  and  $\phi^2$  are the parameterization functions on the two tetrahedra adjacent to  $\mathcal{F}$ .

**Boundary Alignment** For each tetrahedron on the volume boundary, we find the parameterization gradient that is most aligned with the boundary normal, normalize it, and compare it to the normal:

$$Q_{\text{balign}} = \frac{1}{|\mathcal{T}_{\text{bdry}}|} \sum_{\mathcal{T}_{\text{bdry}}} \left( 1 - \max_{i=1}^3 \frac{|\nabla \phi_i \cdot \mathbf{n}|}{\|\nabla \phi_i\|} \right), \quad (21)$$

where  $\mathbf{n}$  is the normal to the tetrahedron's boundary facet. All methods compared in this experiment in principle design their frame fields  $\mathbf{F}$  to align to the volume boundary; however, distortion during global integration can violate boundary alignment.

## 7.2 Choice of Datasets

There are several datasets that previous works on symmetric frame field design evaluate on. In particular, Fang et al. [FHTB23] evaluate on a set of models from the AIM@SHAPE dataset, Liu and Bommes [LB23] evaluate on HexMe, and Palmer et al. [PBS20] evaluate on a collection of models that are commonly otherwise used as test models in geometry processing.

**Sampling the Encyclopedia of Polyhedra** We propose using a dataset of polyhedra as a benchmark for seamless volumetric parameterization. Regular convex polyhedra are on the one hand a canonical class of geometric objects, and on the other pose a surprising level of challenge to state of the art volume parameterization methods (e.g. 6 out of 15 of the “nasty” models in the HexMe dataset [BRK\*22] are convex polyhedra.)

More specifically, we choose as one of our evaluation datasets a collection of 120 convex polyhedra [Ali18] that includes the uniform polyhedra (i.e. the Platonic solids, the Archimedean solids,

and the first few prisms and antiprisms) and the 92 Johnson solids, named after Norman Johnson (the non-uniform convex regular-faced polyhedra whose vertices aren't transitive). This set of 120 examples was curated from a much larger digital repository of polyhedra largely developed by the mathematical artist George Hart [Har96]. We remesh each polyhedron using TetWild [HZG\*18], with a target of 3000 verts for each polyhedron (which translates into approximately 10k tets per model).

In our experience, it can be quite challenging to get parameterization algorithms to pass very simple test cases. We advocate for the use of the 120 polyhedra as a standard parameterization benchmark, despite such models not being representative of any immediate “practical” application, for several reasons:

1. the singular structures of convex polyhedral are easier to reason about than those of more complex shapes;
2. the dataset is large and contains many examples at each of several levels of geometric complexity, so that outlier models are unlikely to skew aggregate statistics;
3. it is easy to mesh all of the polyhedra at a consistent resolution, removing one source of variability between models;
4. a parameterization algorithm that performs poorly on convex polyhedra is unlikely to succeed on CAD models, standard computer graphics test meshes, organic shapes, or other more application-driven datasets.

We ran experiments using this dataset on a cluster of Dell PowerEdge M610 servers with a E5530 Xeon (quad-core) @ 2.4GHz processor. Our reference C++ implementation converged within a day for all examples. In Figure 8 we document some of the results of MINT MESH with integer grid parameterization. In Figures 13, 14 and 15, we perform seamless parameterization on three polyhedra from this test set, in particular, the *bilunabirotunda*, *dodecahedron*, and *square-gyrobicupola* and compare against a number of baseline methods (described below). We plot per-model average quality metrics in Figure 20, and aggregate the mean and standard deviation of these metrics in Table 1.

**Parameterizing HexMe Industrial** We perform a second set of experiments on the HexMe Industrial dataset. To control the computational cost of the experiment, we remesh each model using TetWild [HZG\*18], giving it a target of 3k vertices. (In some cases TetWild was not able to achieve this target, e.g. for *i28u*, a tire model with thin walls, TetWild produced a mesh with 60000 vertices.)

In Figure 9 we illustrate the parameterization results of MINT MESH on a selection of models from this dataset. In Figures 11 and 12, we compare the results with other octahedral baselines, and document per-model statistics in Figure 5.

### 7.3 Choice of Baseline Methods

We compare against three different baseline methods for seamless parameterization via boundary aligned octahedral frame field optimization. These methods generally produce qualitatively similar, but not identical, results. We document these qualitative differences in our comparison figures below. To the extent to which it is possible, we run all the solvers with default parameters. For two of these

implementations, we also benchmark against the method with optional additional features unique to that method turned on. In order to quantitatively evaluate these methods fairly, we take the optimized frame field from each method, renormalize it to have average frame determinant 3375, and pass the resulting field into our implementation of CubeCover [NRP11]. We evaluate the parameterization which results. We perform these experiments both with and without integer grid constraints on the boundary.

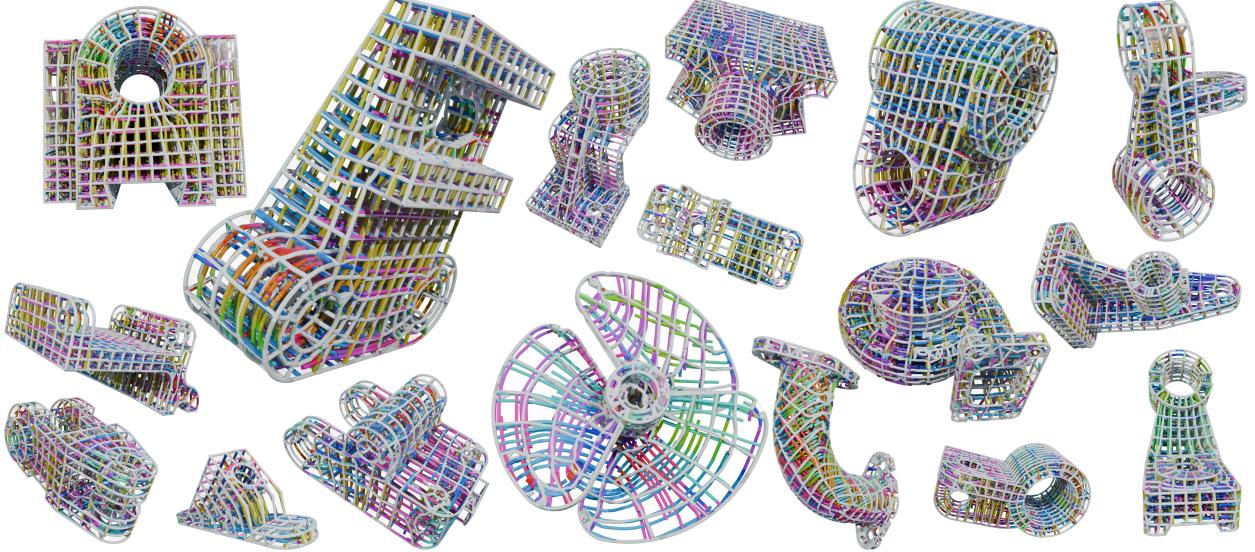
**METRIC GUIDED** [FHTB23] This method first solves for a metric over the volume encoding user-provided constraints like sharp feature alignment, and then uses this metric to optimize an octahedral frame field. We only evaluate the octahedral frame solver, without the additional metric customization pre-processing step, e.g. we use the solver which minimizes the objective specified in their Equation 18 with the metric set to identity. Among baseline implementations, we find this implementation produces the best parameterizations on average with respect to our quality metrics on the polyhedral dataset.

**ARFF** [PBS20] This method supports optimizing both octahedral and anisotropic orthogonal (“odeco”) frame fields. We tested both options, which we label OCTA or ODECO respectively. We use the *mMBO* solver, as Palmer et al. reported that it gave the best practical results in their paper. We find that allowing frames to rescale (the ODECO option) results in higher quality parameterizations on average.

Of the methods we tested, ARFF is the only one that discretizes frame fields on mesh vertices. Our CubeCover implementation operates on tet based fields, and there is no canonical choice for transferring symmetric frames from verts to tets. For each tetrahedron, we locally comb the frame fields at its four corners, average corresponding vectors, and assign this average to the tetrahedron. This approach is simple, but can introduce spurious local artifacts (visible in the figures) that worsen the method's benchmark performance. (Note though that this resampling step will not alter the global singular structure of the frame field.) A different approach to resampling would be to average in the space of spherical harmonic coefficients and then project the result onto  $GL(3)/\mathcal{O}$ , which might remove some of the artifacts visible in our experiments, but it is not clear that this alternative would give good results in the neighborhood of singularities.

**LMFF** [LB23] To our knowledge, this method is the state of the art field-based hexahedral meshing pipeline. The publicly-released source code supports optimizing for either per-vertex and per-tetrahedron frame fields. To avoid the resampling post-processing step mentioned above, we use their tet-based formulation—note that in their publication they report results for their vertex-based solver. As a first step in their pipeline, they initialize fields with an implementation of Ray et al.'s older octahedral frame field optimization method [RSL16], with an additional sharp feature alignment constraint. We call this initialization method LMFF RAY and evaluate it on our benchmark; we find that it produces parameterizations comparable to those produced by METRIC GUIDED.

We also evaluate the fields after applying Liu and Bommes's local meshability optimization step; we call this method LMFF ME-



**Figure 9:** A gallery of some of intgrid parametrizations and dual field streamlines from MINT MESH on the HexMe industrial dataset.

SHABLE in our benchmark. In their pipeline, this step performs additional remeshing as a pre-process, which we did not disable; we also did not tune their crease alignment threshold, and as a consequence some artifacts are visible near sharp creases in some of the LMFF MESHABLE results. We find that this local meshability optimization step makes fields significantly less smooth, and on some examples in the dataset the result is not globally integrable, leading to highly degenerate results from CubeCover (which explains the poor average case performance in Table 1). The plots in Figure 20 reflect that in some cases this method does result in parameterizations which are better than those from LMFF RAY alone. We do not systematically evaluate the additional curl correction step described in Equation 6 of their paper, but tried several examples where this step resulted in frames becoming extremely anisotropic.

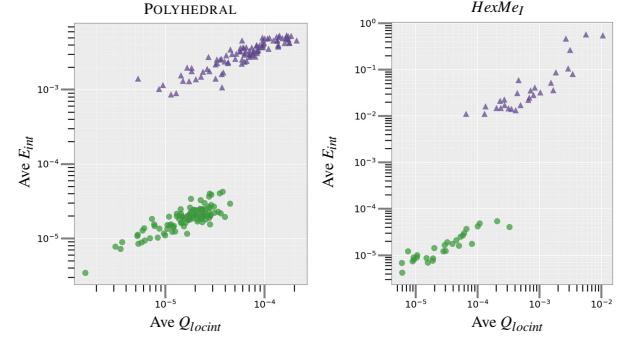
Many of the post-processing steps proposed by Liu and Bommes could be applied generically to other field design methods to assemble a hex meshing pipeline. We believe such a benchmark would be useful for assessing the practical benefits of different field optimization methods on the quality of the final hexahedral mesh, but would require significant engineering effort that we leave for future work.

**MINT3D, associated with this work** We evaluate our reference implementation both with (MINT MESH) and without (MINT OCTA) our local integrability constraint.

#### 7.4 Analysis

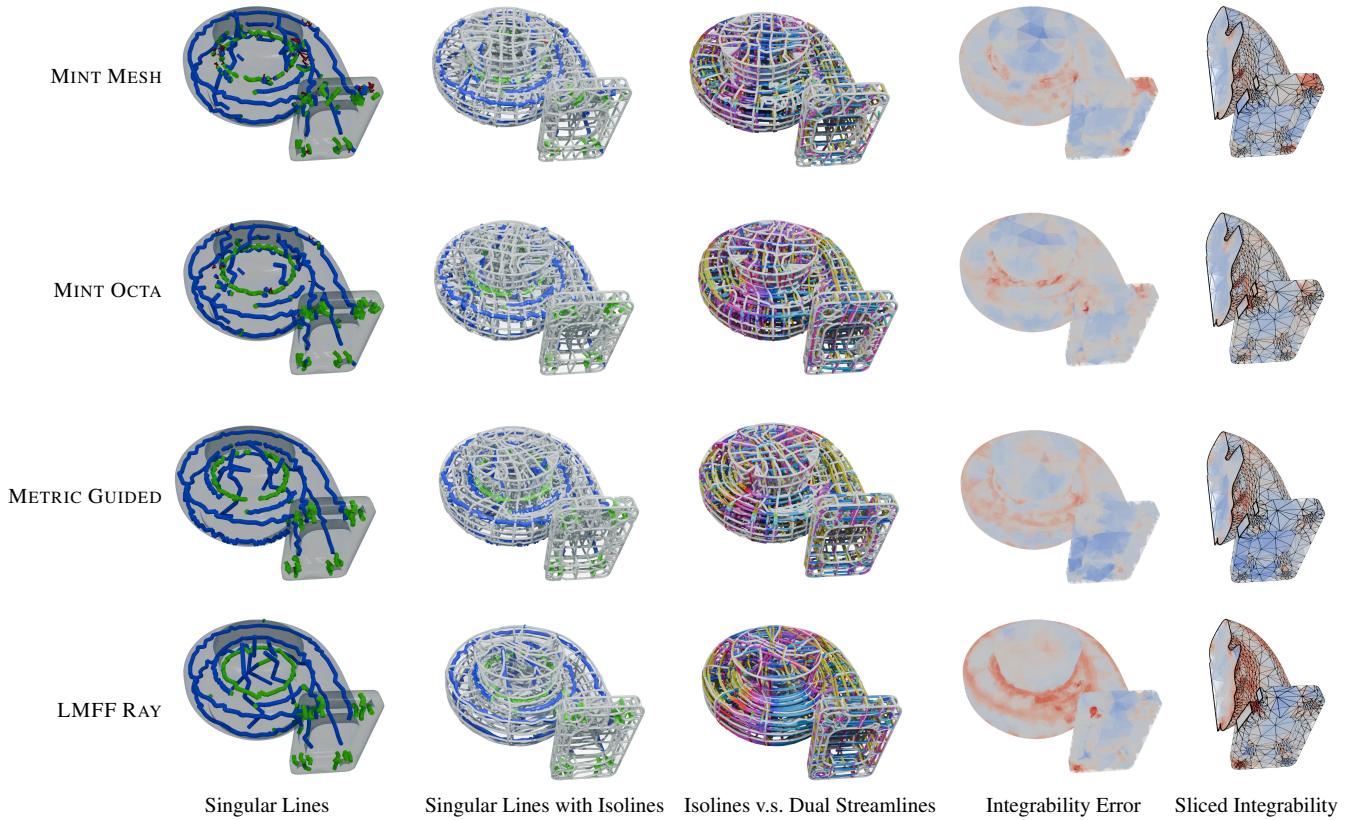
We performed quantitative evaluations on a dataset of polyhedra and a dataset of industrial models.

**Performance on Polyhedral Dataset** We find that our formulation without integrability (MINT OCTA) already outperforms the



**Figure 10:** We show the average relative integrability residual and symmetric integrability residuals on our two test datasets, on a log-log plot. In **purple** we show the results on MINT OCTA and in **green** the results on MINT MESH. When  $E_{int}$  is enforced as a constraint via the penalty method, recovered solutions have significantly lower mean  $E_{int}$  residual, as well as significantly less combed integrability  $Q_{locint}$ . We note also that Eq. 3 was chosen to be quartic, to have the same lowest order scaling behavior as Eq. 5. This experiment demonstrates that  $E_{int}$  is a meaningful proxy for  $Q_{locint}$ . It also demonstrates that our numerical approach consistently reduced the  $E_{int}$  residual by several orders of magnitude in this ablation study.

baseline methods with respect to our parameterization quality metrics. We attribute this improved performance to the extra flexibility to optimize over non-orthogonal and non-unit frame fields directly, unlike previous published approaches to symmetric frame field design. Beyond this, we find that our formulation of symmetric integrability consistently produces fields with lower  $Q_{int}$  than baseline methods.



**Figure 11:** The integer grid parameterization over the optimal fields obtained using different methods. In this case, LMFF RAY failed completely, exhibiting a large integrability error and a noisy parameterization, while the other methods produced reasonable results. Among them, the proposed method (MINT MESH) achieved the smallest integrability residual with well-structured and compatible isolines.

As discussed above, the reported parameterization quality metrics for Palmer et al.’s method [PBS20] are clouded because of noise introduced by the need to resample frames from vertices onto tetrahedra. However something which can already be seen from this evaluation is that the ARFF ODECO method yields improved results with respect to our mesh quality metrics relative to ARFF OCTA, at the cost of fields becoming more anisotropic. This improvement mirrors the behavior that we see where MINT MESH and MINT OCTA outperform the other methods on all of the parameterization quality metrics except for  $Q_{\text{aniso}}$ . These previous methods do not allow frames to rescale during frame field design and so give results that are less anisotropic at the cost of the other quality metrics.

In Figure 10, we evaluate the extent to which our numerical method successfully enforces local integrability as a constraint. The results in Figures 13 and 14 show that our method produces qualitatively different results with and without inclusion of the integrability constraint. The addition of the integrability constraint (MINT MESH) makes the singular structures more symmetric and makes the resulting parameterizations considerably more boundary aligned; in addition, in the MINT MESH results the field streamlines are globally better aligned with the parametric grid. In Figure 15,

we see a dramatic difference between the results of MINT MESH and MINT OCTA, with the latter converging to a somewhat degenerate field poorly aligned with the mesh boundary.

To aggregate data over our dataset, we take METRIC GUIDED as a reference implementation. In Figure 20, we average each parameterization quality statistic over each mesh, sort the meshes in increasing difference of this quality metric between METRIC GUIDED and MINT MESH, and display all of the benchmark methods with respect to this ordering. We also plot this difference between methods as an additional line in the plot. This visualization technique allows one to quickly confirm visually that our method yields a significant improvement on the displayed quality metrics both with respect to the median example in the dataset. In Table 1, we then aggregate the plotted statistics to compute an overall mean and variance over the entire dataset. This table illustrates that MINT MESH does in fact outperform baselines in terms of average performance on all metrics except for parameterization anisotropy.

**Performance on the HexMe<sub>1</sub> Dataset** In general we expect the implementations of METRIC GUIDED and LMFF RAY to produce similar results, as they are both minimizing similar objective functions discretized in the same way, though the latter includes a crease

alignment penalty not present in the former. This is mostly true, but we do also see a statistically significant difference in the results of these two methods with respect to our metrics. In Figure 11 we find that the method of Liu and Bommes [LB23] produces a significantly worse parameterization, whereas in Figure 12 we find that the method of Fang et al. [FHTB23] instead produces a parameterization with the most distortion. In both examples MINT MESH outperforms baselines. Revisiting the aggregate results shown in Figure 5, we note that MINT MESH on average finds frame fields which are more aligned with the resulting parameterizations after CubeCover (with integer grid boundary constraints) and that these parameterizations are on average better than those produced by baseline methods on this dataset.

## 8 Conclusion

In this work, we presented a formulation of local integrability of symmetric frame fields in terms of a symmetric moment representation of the frames. We incorporated these constraints into an “as-octahedral-as-possible” frame field optimization algorithm and showed that enforcing symmetric integrability as a constraint during frame field design for seamless and integer-grid boundary-aligned volume parameterization improves the quality of the resulting parameterization compared to baseline methods.

### 8.1 Future Work

There remains much to be done to further develop the approach of designing symmetric frame fields subject to an integrability constraint.

**Exploiting “Hidden Convexity”** Our formulation uses frame degrees of freedom. This conceptually simplifies the process of deriving additional objective terms like our “as-octahedral-as-possible” fairness energy. Some previous works on symmetric frame field design like that of Palmer et al. [PBS20] take the symmetric tensor coefficients themselves as the optimization DOFs. We leave careful validation of this promising numerical approach (e.g. recasting Eq. 11 as a Riemannian optimization problem [AAMT09]) for future work, but note that applying such an approach would make the problem significantly more convex (e.g.  $E_{\text{int}}$  from Eq. 5 would become a linear constraint, and  $E_{\text{smooth}}$  would be quadratic in these DOFs), at the cost of needing to enforce an extra (non-linear, but local) constraint that the DOFs on each tet live in the image of the lift  $L$ .

**Studying the Octahedral Limit** The frame field design literature has seen success in applying the idea of the “Ginzburg-Landau” functional to meshing problems [VO19]. It would be interesting to explore this idea in the context of symmetric frame field design over the representation for  $GL(3)/\mathcal{O}$  introduced in this work, along the lines of what is explored in [SFCBCV19] in 2D. In particular, it would be interesting to study the fields which result from optimizing the MINT MESH objective in the limit where the octahedral fairness weight goes to infinity. We note that if integrability is enforced while taking this limit, solutions actually correspond to a *different* PDE, often called the Aviles-Giga functional in the numerical analysis literature [Koh06]. Performing such an optimization in a stable way in practice is a non-trivial algorithm design challenge.

**Injective  $GL(3)/\mathcal{O}$  Frame Field Design** One notable limitation of our volumetric parameterization algorithm is that it does not explicitly guarantee that the parameterization is locally injective. In the surface parameterization literature this problem is typically approached by adding a barrier potential to the objective whose energy diverges when frames degenerate [SKPSH13, SS15]. We leave more careful exploration of how best to formulate a barrier potential for this problem to future work, noting also that it may be fruitful to draw on ideas for barrier potentials used in physics simulation (particularly for things like neo-Hookean elasticity [SGK18] and collision response [LFS\*20]).

**Sharp Feature Alignment** A natural consideration which we do not explicitly include in the MINT MESH model is alignment to sharp features. One approach to this would be to explicitly constrain boundary frames to align to features; another would be to add additional metric degrees of freedom to the optimization problem, as advocated for by Fang et al. [FHTB23].

**Globally Meshable Frame Field Design** In this work, we explicitly do *not* study the problem of robust hexahedral mesh generation.

One necessary meshability condition that we do not consider, for example, is that meshable frame fields must only have singular curves of index  $\pm \frac{1}{4}$ , i.e. circulating around a singular curve must only swap two of the frame field labels and not all three. Singular curves of index  $\pm \frac{1}{3}$  or  $\pm \frac{1}{6}$  cannot lie along an integer grid line in the parameter domain.

Nothing about our formulation of local integrability prevents such forbidden singularities from occurring. We do note that, in the smooth setting,  $E_{\text{smooth}}$  already weakly regularizes away such singularities. In particular all frame vectors must go to zero in neighborhoods of integrable index  $\frac{1}{3}$  singularities, whereas only two must go to zero in neighborhoods of index  $\frac{1}{4}$  singularities, and thus the latter in principle have lower smoothness energy in local neighborhoods around singular curves. See Figure 17 for an illustration of the analogous situation in 2D, where the smoothness energy promotes creation of integrable line field singularities. This observation does *NOT* imply that global minimizers of the MINT MESH cannot admit  $\frac{1}{3}$  singularities in the smooth setting, merely that local neighborhoods have higher energy.

Another desiderata identified by Liu and Bommes [LB23] is that a frame field design energy for parameterization should discourage the formation of “zipper nodes”. This is not addressed in the present formulation of MINT MESH.

**Fitting Symmetric Frame Fields to Data** As we mention in the introduction, symmetric tensor fields arise naturally in many disparate applications, from topology optimization to brain imaging. It would be interesting to more carefully consider data from such applications and develop symmetric frame field design objectives for processing real world data on tasks other than volume parameterization. For example, in DTI imaging, it remains an open question how to best perform “global tractography” [JJB11, PYR\*14], and enforcing frame orthogonality may not be the best assumption for obtaining physiologically accurate reconstructions of neural tracts. Exploring this and other application areas in greater detail would be an exciting direction for future work.

## 8.2 Acknowledgements

The authors would like to thank number researchers for their help along the way, including: Albert Chern, Keenan Crane, Paul Kry, Rook Bridson, David Bommes, Peter Schröder, Katia Bertoldi, Yousuf Soliman, Nick Sharp, Oded Stein, Mark Gillespie, Nicole Feng, Marcel Padilla, Oliver Gross, Stephanie Wang, Marco Ravelo, Zhenchen Wang, Matheus da Silva Araujo, Olga Gutăan, Kevin Song, Sushrut Karmalkar, Matt Jordan, Surbhi Goel, Sepideh Maleki, and many others who contributed ideas and support over the course of this quite long running research project. We would in particular like to thank the Summer Geometry Institute (SGI) and the several cohorts of SGI fellows whose research explorations of geometry processing of non-manifold surfaces also informed this project.

## References

- [AAMT09] ABSIL P., ANDREWS B., MAHONY R., TRUMPF J.: All roads lead to newton: Feasible second-order methods for equality-constrained optimization. URL: <https://optimization-online.org/2009/08/2386/>.
- [AG87] AVILES P., GIGA Y.: A mathematical problem related to the physical theory of liquid crystal configurations. In *Miniconference on geometry/partial differential equations* (Canberra, AUS, 1987), vol. 2, Centre for Mathematical Analysis, The Australian National University, pp. 1–16. URL: <https://projecteuclid.org/euclid.pcma/1416336633>, doi:10.1145/3306346.3323043.
- [AJL\*19] ARORA R., JACOBSON A., LANGLOIS T. R., HUANG Y., MUELLER C., MATUSIK W., SHAMIR A., SINGH K., LEVIN D. I. W.: Volumetric Michell trusses for parametric design & fabrication. In *Proceedings of the ACM Symposium on Computational Fabrication* (New York, NY, USA, 2019), SCF ’19, Association for Computing Machinery. URL: <https://doi.org/10.1145/3328939.3328999>.
- [Ali18] ALISON N.: Polyhedra viewer, 2018. URL: <https://polyhedra.tessera.li/>.
- [ATK17] ANDREWS S., TEICHMANN M., KRY P. G.: Geometric stiffness for real-time constrained multibody dynamics. *Comput. Graph. Forum* 36, 2 (May 2017), 235–246. URL: <https://doi.org/10.1111/cgf.13122>, doi:10.1111/cgf.13122.
- [BRK\*22] BEAUFORT P.-A., REBEROL M., KALMYKOV D., LIU H., LEDOUX F., BOMMES D.: Hex me if you can. *Computer Graphics Forum* 41, 5 (2022), 125–134. doi:<https://doi.org/10.1111/cgf.14608>.
- [BTGBD24] BRAUNE T., TONG Y., GAY-BALMAZ F., DESBRUN M.: A discrete exterior calculus of bundle-valued forms, 06 2024. arXiv: 2406.05383, doi:10.48550/arXiv.2406.05383.
- [BZK09] BOMMES D., ZIMMER H., KOBBELT L.: Mixed-integer quadrangulation. In *ACM SIGGRAPH 2009 Papers* (New York, NY, USA, 2009), SIGGRAPH ’09, Association for Computing Machinery. URL: <https://doi.org/10.1145/1576246.1531383>, doi:10.1145/1576246.1531383.
- [CBB\*21] COTTAAR M., BASTIANI M., BODDU N., GLASSER M. F., HABER S., VAN ESSEN D. C., SOTIROPOULOS S. N., JABBDI S.: Modelling white matter in gyral blades as a continuous vector field. *NeuroImage* 227 (2021), 117693. doi:<https://doi.org/10.1016/j.neuroimage.2020.117693>.
- [CC19] CORMAN E., CRANE K.: Symmetric moving frames. *ACM Trans. Graph.* 38, 4 (July 2019). URL: <https://doi.org/10.1145/3306346.3323029>.
- [CCR24] COUPLET M., CHEMIN A., REMACLE J.-F.: *Integrable Frame Fields using Odeco Tensors*. 2024, pp. 53–65. URL: <https://epubs.siam.org/doi/abs/10.1137/1.9781611978001.5>, doi:10.1137/1.9781611978001.5.
- [CDG02] CANTARELLA J., DETURCK D., GLUCK H.: Vector calculus and the topology of domains in 3-space. *The American Mathematical Monthly* 109, 5 (2002), 409–442. URL: <http://www.jstor.org/stable/2695643>.
- [CHRS19] CHEMIN A., HENROTTE F., REMACLE J.-F., SCHAFTINGEN J. V.: *Representing Three-Dimensional Cross Fields Using Fourth Order Tensors*. Springer International Publishing, Cham, 2019, pp. 89–108. URL: [https://doi.org/10.1007/978-3-030-13992-6\\_6](https://doi.org/10.1007/978-3-030-13992-6_6), doi:10.1007/978-3-030-13992-6\_6.
- [CIE\*16] CAMPEN M., IBING M., EBKE H.-C., ZORIN D., KOBBELT L.: Scale-invariant directional alignment of surface parametrizations. In *Proceedings of the Symposium on Geometry Processing* (Goslar, DEU, 2016), SGP ’16, Eurographics Association, p. 1–10. URL: <https://doi.org/10.1111/cgf.12958>, doi:10.1111/cgf.12958.
- [CLL\*24] CHEN H., LIU H.-T. D., LEVIN D. I., ZHENG C., JACOBSON A.: Stabler neo-hookean simulation: Absolute eigenvalue filtering for projected newton. In *ACM SIGGRAPH 2024 Conference Papers* (New York, NY, USA, 2024), SIGGRAPH ’24, Association for Computing Machinery. URL: <https://doi.org/10.1145/3641519.3657433>, doi:10.1145/3641519.3657433.
- [Cou24] COUPLET M.: Integrable frame fields for quadrilateral and hexahedral meshing, 2024. URL: <http://hdl.handle.net/2078.1/292282>.
- [CPS15] CHERN A., PINKALL U., SCHRÖDER P.: Close-to-conformal deformations of volumes. *ACM Transactions on Graphics (TOG)* 34 (2015), 1 – 13. URL: <https://api.semanticscholar.org/CorpusID:8126899>.
- [DCOC\*21] DESOBRY D., COUDERT-OSMONT Y., CORMAN E., RAY N., SOKOLOV D.: Designing 2d and 3d non-orthogonal frame fields. *Computer-Aided Design* 139 (2021), 103081. doi:<https://doi.org/10.1016/j.cad.2021.103081>.
- [dGLB\*14] DE GOES F., LIU B., BUDNINSKIY M., TONG Y., DESBRUN M.: Discrete 2-tensor fields on triangulations. *Computer Graphics Forum* 33, 5 (2014), 13–24. doi:<https://doi.org/10.1111/cgf.12427>.
- [DHL09] DEMIRALP C., HUGHES J. F., LAIDLAW D. H.: Coloring 3d line fields using boy’s real projective plane immersion. *IEEE Transactions on Visualization and Computer Graphics* 15, 6 (nov 2009), 1457–1464. URL: <https://doi.org/10.1109/TVCG.2009.125>.
- [DVPSH14] DIAMANTI O., VAXMAN A., PANZZO D., SORKINE-HORNUNG O.: Designing n-polyvector fields with complex polynomials. *Computer Graphics Forum* 33, 5 (2014), 1–11. doi:<https://doi.org/10.1111/cgf.12426>.
- [DVPSH15] DIAMANTI O., VAXMAN A., PANZZO D., SORKINE-HORNUNG O.: Integrable polyvector fields. *ACM Trans. Graph.* 34, 4 (jul 2015). URL: <https://doi.org/10.1145/2766906>, doi:10.1145/2766906.
- [Eri14] ERICKSON J.: Efficiently hex-meshing things with topology. *Discrete Comput. Geom.* 52, 3 (oct 2014), 427–449. URL: <https://doi.org/10.1007/s00454-014-9624-3>, doi:10.1007/s00454-014-9624-3.
- [FHTB23] FANG X., HUANG J., TONG Y., BAO H.: Metric-driven 3d frame field generation. *IEEE Transactions on Visualization and Computer Graphics* 29, 4 (apr 2023), 1964–1976. URL: <https://doi.org/10.1109/TVCG.2021.3136199>.
- [FTd\*14] FRIEDMAN J. I., TANG C. Y., DE HAAS H. J., CHANGCHIEN L., GOLIASCH G., DABAS P., WANG V., FAYAD Z. A., FUSTER V., NARULA J.: Brain imaging changes associated with risk factors for cardiovascular and cerebrovascular disease in asymptomatic patients. *JACC: Cardiovascular Imaging* 7, 10 (2014), 1039–1053. URL: <https://www.sciencedirect.com/science/article/pii/S1936878X14005488>, doi:10.1016/j.jcmg.2014.06.014.

- [GHX<sup>\*</sup>17] GAO X., HUANG J., XU K., PAN Z., DENG Z., CHEN G.: Evaluating Hex-mesh Quality Metrics via Correlation Analysis. *Computer Graphics Forum* (2017). doi:[10.1111/cgf.13249](https://doi.org/10.1111/cgf.13249).
- [GJTP17] GAO X., JAKOB W., TARINI M., PANZZO D.: Robust hex-dominant mesh generation using field-guided polyhedral agglomeration. *ACM Trans. Graph.* 36, 4 (jul 2017). URL: <https://doi.org/10.1145/3072959.3073676>.
- [Har96] HART G.: Virtual polyhedra, 1996. URL: <https://georgehart.com/virtual-polyhedra/vp.html>.
- [HTWB11] HUANG J., TONG Y., WEI H., BAO H.: Boundary aligned smooth 3d cross-frame field. *ACM Trans. Graph.* 30, 6 (dec 2011), 1–8. URL: <https://doi.org/10.1145/2070781.2024177>.
- [HZG<sup>\*</sup>18] HU Y., ZHOU Q., GAO X., JACOBSON A., ZORIN D., PANZZO D.: Tetrahedral meshing in the wild. *ACM Trans. Graph.* 37, 4 (July 2018), 60:1–60:14. URL: <http://doi.acm.org/10.1145/3197517.3201353>, doi:[10.1145/3197517.3201353](https://doi.acm.org/10.1145/3197517.3201353).
- [Jac] JACOBSON A.: How can one construct a sparse null space basis using recursive lu decomposition? MathOverflow. URL:<https://mathoverflow.net/q/253995> (version: 2016-11-05). URL: <https://mathoverflow.net/q/253995>, arXiv:<https://mathoverflow.net/q/253995>.
- [JJB11] JBABDI S., JOHANSEN-BERG H.: Tractography: Where do we go from here? *Brain Connectivity* 1, 3 (2011), 169–183. PMID: 2243046. URL: <https://doi.org/10.1089/brain.2011.0033>, doi:[10.1089/brain.2011.0033](https://doi.org/10.1089/brain.2011.0033).
- [KCPS15] KNÖPPEL F., CRANE K., PINKALL U., SCHRÖDER P.: Stripe patterns on surfaces. *ACM Trans. Graph.* 34, 4 (July 2015). URL: <https://doi.org/10.1145/2767000>.
- [KNP07] KÄLBERER F., NIESER M., POLTHIER K.: Quadcover - surface parameterization using branched coverings. *Computer Graphics Forum* 26, 3 (2007), 375–384. doi:<https://doi.org/10.1111/j.1467-8659.2007.01060.x>.
- [Koh06] KOHN R.: Energy-driven pattern formation. pp. 359–383. 25th International Congress of Mathematicians, ICM 2006 ; Conference date: 22-08-2006 Through 30-08-2006.
- [KS23] KIPPING J., SCHÜPPSTUHL T.: Load-oriented nonplanar additive manufacturing method for optimized continuous carbon fiber parts. *Materials* 16, 3 (2023). URL: <https://www.mdpi.com/1996-1944/16/3/998>, doi:[10.3390/ma16030998](https://doi.org/10.3390/ma16030998).
- [LB23] LIU H., BOMMES D.: Locally meshable frame fields. *ACM Transactions on Graphics* 42, 4 (2023). doi:[10.1145/3592457](https://doi.org/10.1145/3592457).
- [LBK16] LYON M., BOMMES D., KOBBELT L.: HexEx: Robust hexahedral mesh extraction. *ACM Trans. Graph.* (2016). URL: <http://dx.doi.org/10.1145/2897824.2925976>, doi:[10.1145/2897824.2925976](https://doi.org/10.1145/2897824.2925976).
- [LCD<sup>\*</sup>09] LENGLER C., CAMPBELL J., DESCOTEAUX M., HARO G., SAVADJIEV P., WASSERMANN D., ANWANDER A., DERICHÉ R., PIKE G., SAPIRO G., SIDDIQI K., THOMPSON P.: Mathematical methods for diffusion mri processing. *NeuroImage* 45, 1, Supplement 1 (2009), S111–S122. Mathematics in Brain Imaging. URL: <https://www.sciencedirect.com/science/article/pii/S105381190801197X>.
- [LFS<sup>\*</sup>20] LI M., FERGUSON Z., SCHNEIDER T., LANGLOIS T., ZORIN D., PANZZO D., JIANG C., KAUFMAN D. M.: Incremental potential contact: intersection-and inversion-free, large-deformation dynamics. *ACM Trans. Graph.* 39, 4 (Aug. 2020). URL: <https://doi.org/10.1145/3386569.3392425>.
- [LLFF<sup>\*</sup>23] LONGVA A., LÖSCHNER F., FERNÁNDEZ-FERNÁNDEZ J. A., LARIONOV E., ASCHER U. M., BENDER J.: Pitfalls of projection: A study of newton-type solvers for incremental potentials. *ArXiv abs/2311.14526* (2023). URL: <https://api.semanticscholar.org/CorpusID:265445656>.
- [LLX<sup>\*</sup>12] LI Y., LIU Y., XU W., WANG W., GUO B.: All-hex meshing using singularity-restricted field. *ACM Trans. Graph.* 31, 6 (nov 2012). URL: <https://doi.org/10.1145/2366145.2366196>.
- [LZC<sup>\*</sup>18] LIU H., ZHANG P., CHIEN E., SOLOMON J., BOMMES D.: Singularity-constrained octahedral fields for hexahedral meshing. *ACM Trans. Graph.* 37, 4 (July 2018). URL: <https://doi.org/10.1145/3197517.3201344>.
- [MTG<sup>\*</sup>11] MARTIN S., THOMASZEWSKI B., GRINSPUN E., GROSS M. H., ZURICH E.: Example-based elastic materials. *ACM SIGGRAPH 2011 papers* (2011). URL: <https://api.semanticscholar.org/CorpusID:261462978>.
- [Noc06] *Line Search Methods*. Springer New York, New York, NY, 2006, pp. 30–65. URL: [https://doi.org/10.1007/978-0-387-40065-5\\_3](https://doi.org/10.1007/978-0-387-40065-5_3), doi:[10.1007/978-0-387-40065-5\\_3](https://doi.org/10.1007/978-0-387-40065-5_3).
- [NRP11] NIESER M., REITEBUCH U., POLTHIER K.: Cubecover-parameterization of 3d volumes. *Computer Graphics Forum* 30, 5 (2011), 1397–1406. doi:<https://doi.org/10.1111/j.1467-8659.2011.02014.x>.
- [NWZ18] NOCHETTO R. H., WALKER S. W., ZHANG W.: The ericksen model of liquid crystals with colloidal and electric effects. *Journal of Computational Physics* 352 (2018), 568–601. doi:<https://doi.org/10.1016/j.jcp.2017.09.035>.
- [PBS20] PALMER D., BOMMES D., SOLOMON J.: Algebraic representations for volumetric frame fields. *ACM Trans. Graph.* 39, 2 (Apr. 2020). URL: <https://doi.org/10.1145/3366786>.
- [PCS<sup>\*</sup>22] PIETRONI N., CAMPEN M., SHEFFER A., CHERCHI G., BOMMES D., GAO X., SCATENI R., LEDOUX F., REMACLE J., LIVESU M.: Hex-mesh generation and processing: A survey. *ACM Trans. Graph.* 42, 2 (oct 2022). URL: <https://doi.org/10.1145/3554920>.
- [PP03] POLTHIER K., PREUSS E.: Identifying vector field singularities using a discrete hodge decomposition. In *Visualization and Mathematics III* (Berlin, Heidelberg, 2003), Hege H.-C., Polthier K., (Eds.), Springer Berlin Heidelberg, pp. 113–134. doi:[10.1007/978-3-662-05105-4\\_6](https://doi.org/10.1007/978-3-662-05105-4_6).
- [PPicv<sup>\*</sup>19] POLLARD J., POSNJAK G., ČOPAR S., MUŠEVIĆ I., ALEXANDER G. P.: Point defects, topological chirality, and singularity theory in cholesteric liquid-crystal droplets. *Phys. Rev. X* 9 (Apr 2019), 021004. doi:[10.1103/PhysRevX.9.021004](https://doi.org/10.1103/PhysRevX.9.021004).
- [PPTSH14] PANZZO D., PUPPO E., TARINI M., SORKINE-HORNUNG O.: Frame fields: anisotropic and non-orthogonal cross fields. *ACM Trans. Graph.* 33, 4 (jul 2014). URL: <https://doi.org/10.1145/2601097.2601179>, doi:[10.1145/2601097.2601179](https://doi.org/10.1145/2601097.2601179).
- [PRK<sup>\*</sup>17] PALACIOS J., ROY L., KUMAR P., HSU C.-Y., CHEN W., MA C., WEI L.-Y., ZHANG E.: Tensor field design in volumes. *ACM Trans. Graph.* 36, 6 (Nov 2017). doi:[10.1145/3130800.3130844](https://doi.org/10.1145/3130800.3130844).
- [PYR<sup>\*</sup>14] PESTILLI F., YEATMAN J. D., ROKEM A., KAY K. N., WANDELL B. A.: Evaluation and statistical inference for human connectomes. *Nature methods* 11, 10 (October 2014), 1058–1063. URL: <https://europepmc.org/articles/PMC4180802>, doi:[10.1038/nmeth.3098](https://doi.org/10.1038/nmeth.3098).
- [RLL<sup>\*</sup>06] RAY N., LI W. C., LÉVY B., SHEFFER A., ALLIEZ P.: Periodic global parameterization. *ACM Trans. Graph.* 25, 4 (oct 2006), 1460–1485. URL: <https://doi.org/10.1145/1183287.1183297>, doi:[10.1145/1183287.1183297](https://doi.org/10.1145/1183287.1183297).
- [RPPG19] RAZAFINDRAZAKA F. H., POELKE K., POLTHIER K., GOUBERGRITS L.: A consistent discrete 3d hodge-type decomposition: implementation and practical evaluation, 2019. arXiv:[1911.12173](https://arxiv.org/abs/1911.12173).
- [RSL16] RAY N., SOKOLOV D., LÉVY B.: Practical 3d frame field generation. *ACM Trans. Graph.* 35, 6 (nov 2016). URL: <https://doi.org/10.1145/2980179.2982408>.
- [SFCBCV19] SAGEMAN-FURNAS A. O., CHERN A., BEN-CHEN M., VAXMAN A.: Chebyshev nets from commuting polyvector fields. *ACM Trans. Graph.* 38, 6 (nov 2019). doi:[10.1145/3355089.3356564](https://doi.org/10.1145/3355089.3356564).

- [SGK18] SMITH B., GOES F. D., KIM T.: Stable neo-hookean flesh simulation. *ACM Trans. Graph.* 37, 2 (Mar. 2018). URL: <https://doi.org/10.1145/3180491>.
- [SKPSH13] SCHÜLLER C., KAVAN L., PANZOZZO D., SORKINE-HORNUNG O.: Locally injective mappings. *Computer Graphics Forum* 32, 5 (2013), 125–135. doi:<https://doi.org/10.1111/cgf.12179>.
- [SS15] SMITH J., SCHAEFER S.: Bijective parameterization with free boundaries. *ACM Trans. Graph.* 34, 4 (July 2015). URL: <https://doi.org/10.1145/2766947>.
- [SVB17] SOLOMON J., VAXMAN A., BOMMES D.: Boundary element octahedral fields in volumes. *ACM Trans. Graph.* 36, 4 (may 2017). URL: <https://doi.org/10.1145/3072959.3065254>.
- [TLHD03] TONG Y., LOMBEYDA S., HIRANI A. N., DESBRUN M.: Discrete multiscale vector field decomposition. *ACM Trans. Graph.* 22, 3 (July 2003), 445–452. URL: <https://doi.org/10.1145/882262.882290>.
- [TSIF05] TERAN J., SIFAKIS E., IRVING G., FEDIKI R.: Robust quasistatic finite elements and flesh simulation. In *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (New York, NY, USA, 2005), SCA ’05, Association for Computing Machinery, p. 181–190. URL: <https://doi.org/10.1145/1073368.1073394>, doi:[10.1145/1073368.1073394](https://doi.org/10.1145/1073368.1073394).
- [VCD\*16] VAXMAN A., CAMPEN M., DIAMANTI O., PANZOZZO D., BOMMES D., HILDEBRANDT K., BEN-CHEN M.: Directional field synthesis, design, and processing. *Computer Graphics Forum* 35 (05 2016), 545–572. doi:[10.1111/cgf.12864](https://doi.org/10.1111/cgf.12864).
- [VO19] VIERTEL R., OSTING B.: An approach to quad meshing based on harmonic cross-valued maps and the ginzburg–landau theory. *SIAM Journal on Scientific Computing* 41, 1 (2019), A452–A479. URL: <https://doi.org/10.1137/17M1142703>.
- [VS09] VYAS V., SHIMADA K.: Tensor-guided hex-dominant mesh generation with targeted all-hex regions. In *Proceedings of the 18th International Meshing Roundtable* (Berlin, Heidelberg, 2009), Clark B. W. (Ed.), Springer Berlin Heidelberg, pp. 377–396. doi:[10.1007/978-3-642-04319-2\\_22](https://doi.org/10.1007/978-3-642-04319-2_22).
- [VZF\*19] VEKHTER J., ZHUO J., FANDINO L. F. G., HUANG Q., VOUGA E.: Weaving geodesic foliations. *ACM Trans. Graph.* 38, 4 (July 2019). URL: <https://doi.org/10.1145/3306346.3323043>.
- [WPS14] WEISSMANN S., PINKALL U., SCHRÖDER P.: Smoke rings from smoke. *ACM Trans. Graph.* 33, 4 (July 2014). doi:[10.1145/2601097.2601171](https://doi.org/10.1145/2601097.2601171).
- [ZCFM23] ZHANG P., CHIANG J. H.-H., FAN X. C., MUNDILOVA K.: Local decomposition of hexahedral singular nodes into singular curves. *Comput. Aided Des.* 158, C (may 2023). URL: <https://doi.org/10.1016/j.cad.2023.103484>, doi:[10.1016/j.cad.2023.103484](https://doi.org/10.1016/j.cad.2023.103484).
- [ZDWT19] ZHAO R., DESBRUN M., WEI G.-W., TONG Y.: 3d hodge decompositions of edge- and face-based vector fields. *ACM Trans. Graph.* 38, 6 (nov 2019). URL: <https://doi.org/10.1145/3355089.3356546>, doi:[10.1145/3355089.3356546](https://doi.org/10.1145/3355089.3356546).

## A Space of Moments

The map  $L: \mathbb{R}^{3 \times 3} \rightarrow \mathcal{M}$  clearly projects to a well-defined function on  $\mathbb{R}^{3 \times 3}/\mathcal{O}$ , since  $L(\mathbf{F}g) = L(\mathbf{F})$  for all  $g \in \mathcal{O}$ . In this section, we prove that this function is injective: that a symmetric frame can be uniquely recovered from its second, fourth, and sixth moments.

The workhorse of our proof is the following lemma:

**Lemma A.1.** *For any  $X, Y, Z \in \mathbb{R}$ , every solution  $(x, y, z)$  to the following system of equations:*

$$x^3 + y^3 + z^3 = X^3 + Y^3 + Z^3 \quad (22)$$

$$x^2 + y^2 + z^2 = X^2 + Y^2 + Z^2 \quad (23)$$

$$x + y + z = X + Y + Z \quad (24)$$

*is one of the six permutations of  $(X, Y, Z)$ .*

*Proof.* Note that by Bézout’s theorem, we should expect that the six solutions given by the permutations of  $(X, Y, Z)$  account for all solutions of the system. But we can show this is true by direct computation: note that

$$\begin{aligned} 2(y^3 + z^3) &= (y+z)(2y^2 - 2yz + 2z^2) \\ &= (y+z) \left[ 3(y^2 + z^2) - (y+z)^2 \right]. \end{aligned}$$

This identity, together with Equations (23) and (24), can be used to eliminate  $y$  and  $z$  from Equation (22), yielding

$$3x^3 - 3(X+Y+Z)x^2 + 3(XY+XZ+YZ)x - 3XYZ = 0$$

which factors as

$$3(x-X)(x-Y)(x-Z) = 0.$$

Assuming without loss of generality that  $x = X$  and eliminating  $z$  from Equation (23) gives  $(y, z) = (Y, Z)$  or  $(Z, Y)$ .  $\square$

Now let  $\mathbf{F} = \{\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3\}$  and  $\mathbf{G} = \{\mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3\}$  be two frames with  $\mathbf{M}_k = L_k(\mathbf{F})$  and  $\mathbf{N}_k = L_k(\mathbf{G})$  their  $k$ th moments. We will prove that if  $\mathbf{M}_k = \mathbf{N}_k$  for  $k = 2, 4, 6$ , then  $\mathbf{F} \sim \mathbf{G}$ .

## Frame Vectors Are Parallel

We begin with the following observation: if  $\mathbf{M}_6 = \mathbf{N}_6$ , then each vector  $\mathbf{g}_j$  is a rescaling of one of the vectors  $\mathbf{f}_i$ . Suppose, for contradiction, that this isn’t true for some  $\mathbf{g}_a$ . Then  $\mathbf{g}_a \times \mathbf{f}_i \neq \mathbf{0}$  for all non-zero vectors  $\mathbf{f}_i \in \mathbf{F}$ ; define

$$\mathbf{f}_i^\perp = \begin{cases} \mathbf{f}_i \times (\mathbf{g}_a \times \mathbf{f}_i), & \mathbf{f}_i \neq \mathbf{0} \\ \mathbf{g}_a, & \mathbf{f}_i = \mathbf{0}. \end{cases}$$

Observe that for all  $i$ ,  $\mathbf{f}_i \cdot \mathbf{f}_i^\perp = 0$ ; moreover  $\mathbf{g}_a \cdot \mathbf{f}_i^\perp > 0$ , since by hypothesis  $\mathbf{g}_a \neq \mathbf{0}$  and in the first case

$$\mathbf{g}_a \cdot [\mathbf{f}_i \times (\mathbf{g}_a \times \mathbf{f}_i)] = \|\mathbf{g}_a \times \mathbf{f}_i\|^2 > 0$$

by invariance of triple products under cyclic permutation.

Now if  $\mathbf{M}_6 = \mathbf{N}_6$ ,

$$\mathbf{M}_6(\mathbf{f}_1^\perp, \mathbf{f}_1^\perp, \mathbf{f}_2^\perp, \mathbf{f}_2^\perp, \mathbf{f}_3^\perp, \mathbf{f}_3^\perp) = \mathbf{N}_6(\mathbf{f}_1^\perp, \mathbf{f}_1^\perp, \mathbf{f}_2^\perp, \mathbf{f}_2^\perp, \mathbf{f}_3^\perp, \mathbf{f}_3^\perp)$$

$$\sum_{i=1}^3 \prod_{j=1}^3 (\mathbf{f}_i \cdot \mathbf{f}_j^\perp)^2 = \sum_{i=1}^3 \prod_{j=1}^3 (\mathbf{g}_i \cdot \mathbf{f}_j^\perp)^2.$$

We take the notation in the first line to mean contracting  $\mathbf{M}_6$  and  $\mathbf{N}_6$  with the 6 given vectors. In the second line, we expand out the definition of  $\mathbf{M}_6$  and simplify the contraction to a sum of products. Since every term in the left-hand sum vanishes, and every term in the right-hand sum is clearly non-negative,

$$0 = \sum_{i=1}^3 \prod_{j=1}^3 (\mathbf{g}_i \cdot \mathbf{f}_j^\perp)^2 \geq \prod_{j=1}^3 (\mathbf{g}_a \cdot \mathbf{f}_j^\perp)^2 > 0,$$

a contradiction.

### Injectivity of $L$

We've shown that every vector in  $\mathbf{G}$  is a rescaling of a vector in  $\mathbf{F}$ ; but we still need to prove that the vectors in the two frames are in one-to-one correspondence with each other with scale factors  $\pm 1$ .

Suppose  $\mathbf{F}$  contains a nonzero vector; otherwise trivially  $\mathbf{F} = \mathbf{G} = \mathbf{0}$ . Without loss of generality we may set it to be  $\mathbf{f}_1$ . Suppose further that there are  $p$  total vectors in  $\mathbf{F}$  parallel to  $\mathbf{f}_1$ ; we may set them to be  $\mathbf{f}_1, \dots, \mathbf{f}_p$ , with the remainder non-zero vectors  $\mathbf{f}_{i>p}$  with  $\mathbf{f}_1 \cdot \mathbf{f}_i^\perp \neq 0$ , where

$$\mathbf{f}_i^\perp = \mathbf{f}_i \times (\mathbf{f}_1 \times \mathbf{f}_i).$$

Moreover set without loss of generality  $\mathbf{g}_1, \dots, \mathbf{g}_q$  to be the vectors in  $\mathbf{G}$  that are scalar multiples of  $\mathbf{f}_1$ ; by the above observation each of the rest must be rescalings of one of the other vectors  $\mathbf{f}_{i>p}$ . For the parallel vectors let  $\mathbf{f}_i = X_i \mathbf{f}_1$  and  $\mathbf{g}_i = x_i \mathbf{f}_1$ . Then since  $\mathbf{M}_k = \mathbf{N}_k$ ,

$$\mathbf{M}_k(\mathbf{f}_{p+1}^\perp, \dots, \mathbf{f}_3^\perp, \mathbf{f}_1, \dots, \mathbf{f}_1) = \mathbf{N}_k(\mathbf{f}_{p+1}^\perp, \dots, \mathbf{f}_3^\perp, \mathbf{f}_1, \dots, \mathbf{f}_1).$$

The left-hand side simplifies to

$$\begin{aligned} \mathbf{M}_k(\mathbf{f}_{p+1}^\perp, \dots, \mathbf{f}_3^\perp, \mathbf{f}_1, \dots, \mathbf{f}_1) &= \\ \sum_{i=1}^p (\mathbf{f}_i \cdot \mathbf{f}_1)^{k-(3-p)} \prod_{j=p+1}^3 (\mathbf{f}_i \cdot \mathbf{f}_j^\perp) &= \\ (\|\mathbf{f}_1\|^{2(k+p-3)} \prod_{j=p+1}^3 (\mathbf{f}_1 \cdot \mathbf{f}_j^\perp)) \sum_{i=1}^p X_i^k & \end{aligned}$$

and the right-hand side to

$$\begin{aligned} \mathbf{N}_k(\mathbf{f}_{p+1}^\perp, \dots, \mathbf{f}_3^\perp, \mathbf{f}_1, \dots, \mathbf{f}_1) &= \\ \sum_{i=1}^q (\mathbf{g}_i \cdot \mathbf{f}_1)^{k-(3-p)} \prod_{j=p+1}^3 (\mathbf{g}_i \cdot \mathbf{f}_j^\perp) &= \\ (\|\mathbf{f}_1\|^{2(k+p-3)} \prod_{j=p+1}^3 (\mathbf{f}_1 \cdot \mathbf{f}_j^\perp)) \sum_{i=1}^q x_i^k. & \end{aligned}$$

Since these expressions must be equal for  $k = 2, 4, 6$  the following system of equations relates the  $X_i$  to the  $x_j$ :

$$\sum_{i=1}^q x_i^6 = \sum_{i=1}^p X_i^6, \sum_{i=1}^q x_i^4 = \sum_{i=1}^p X_i^4, \sum_{i=1}^q x_i^2 = \sum_{i=1}^p X_i^2.$$

Therefore by Lemma A.1 the nonzero  $x_i^2$  must be in bijection with the nonzero  $X_i^2$ , and so every vector parallel to  $\mathbf{f}_1$  in  $\mathbf{F}$  must appear, possibly with a sign flip, in  $\mathbf{G}$  (and there can be no other non-zero

vectors parallel to  $\mathbf{f}_1$  in  $\mathbf{G}$ ). Since the choice of  $\mathbf{f}_1$  was arbitrary, this argument shows that  $\mathbf{F} = \mathbf{G}$  for some  $g \in \mathcal{O}$ .

### Necessity of Three Moments

We've shown that  $L_2(\mathbf{F})$ ,  $L_4(\mathbf{F})$ , and  $L_6(\mathbf{F})$  are sufficient to uniquely recover  $\mathbf{F}$  up to octahedral symmetry. But maybe  $L_6(\mathbf{F})$  isn't actually necessary, and the lower-order moments suffice? This turns out to be wishful thinking. Below, we demonstrate a pair of frames  $\mathbf{F} \not\sim \mathbf{G}$  but with  $L_2(\mathbf{F}) = L_2(\mathbf{G})$  and  $L_4(\mathbf{F}) = L_4(\mathbf{G})$ :

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} \frac{2}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{3}} & \frac{2}{\sqrt{3}} & \frac{-1}{\sqrt{3}} \\ 0 & 0 & 0 \end{bmatrix}. \quad (25)$$

### B Smoothness Operators for Symmetric Frames

It is *not* a priori obvious how to define smoothness of frame fields in  $GL(3)/\mathcal{O}$ . Equation 25 demonstrates that a reasonable smoothness energy must take into account all three tensors in  $L(\mathbf{F})$ ; there exist pairs of frames  $\mathbf{F}, \mathbf{G}$  which are geometrically far from each other but with  $L_k(\mathbf{F}) = L_k(\mathbf{G})$  for  $k = 2$  and 4.

An ideal smoothness energy density would agree with the Dirichlet energy of the combed vector fields  $\mathbf{c}_i$ ,

$$\sum_{i=1}^3 \|\nabla \mathbf{c}_i\|^2,$$

on each neighborhood where a unique local combing exists; but it is not clear that any function of the moment tensors has this property, or to what extent this condition must be relaxed to be achievable.

**Higher Order Bochner Laplacian** Analogous to the Dirichlet energy for vector fields, in the continuous setting, we define our smoothness energy as:

$$E_{\text{bochner}} = \sum_{k=1}^3 \int_{\Omega} \|\nabla(f^{\otimes 2k})\|^2 \quad (26)$$

whose discretization is given in Eq. 6. We find that this formulation of smoothness is simple and effective in practice.

One troubling property of this formulation is that it is not scale equivariant (i.e. the  $k^{\text{th}}$  order tensors scale with the vector norm to the  $k^{\text{th}}$  power), and so minima of this energy may depend on global rescaling. Possible modifications to address this limitation include: taking roots of the higher order factors to ensure quadratic scaling, reweighting lower-order terms to match the highest-order scaling, or leveraging the “Kruskal” representation of tensors, where each rank-1 factor scales with the vector norm rather than its tensor power, akin to the method used by Palmer et al. [PBS20].

To guide future research on frame field smoothness formulations, we propose three unit tests (illustrated in Figure 18 and in the inset figure) where the global minimizer of smoothness, subject to boundary alignment of the frames, is unambiguous. Minimizing any reasonable definition of frame field smoothness from a large number



of random initializations should converge to this global minimizer (and for the first two should ideally produce a solution with zero residual):

1. a cube; where the obvious global smoothness minimizer is a constant frame field aligned with the cube facets;
2. a parallelopiped, where the smoothness minimizer is also constant and aligned with the boundary facets.
3. a sphere; where the minimizer can be globally combed into a radial vector field and two constant vector fields (of arbitrary orientation).

$E_{\text{bochner}}$  was the only approach that we tried that could pass these tests, but the lack of scale equivariance is a limitation whose resolution we leave for future work.

### Beginnings of an Exterior Algebra for Symmetric Frames

Another natural discrete operator that can be defined in a manner similar to the above smoothness operator is symmetric divergence, where a symmetric  $\text{div} = 0$  penalty can be formulated by setting  $\mathbf{e} = \mathcal{F}^\perp$  (the face normal) in equation 5:

$$E_{\text{div}} = \sum_{\mathcal{F}^\perp} \sum_{k=1}^3 \|L_{2k}(\bar{\mathbf{F}}) - L_{2k}(\bar{\mathbf{G}})\|_F^2, \quad (27)$$

where in this case the restriction of the tensors is to the 1D line spanned by the face normal. We note that this operator may be of independent interest in certain applications, for instance in DTI [CBB\*21], and it would be interesting to build a more complete theory of discrete differential operators over *symmetric* frame bundles (e.g. by following ideas from work on regular vector bundles [BTGBD24]), but we leave careful development of these ideas to future work.

### C MINT MESH: Algorithmic Notes

Here we provide additional details regarding the implementation of the numerical algorithm which we evaluate in this paper.

**Ensuring Problem Feasibility** In our reference implementation, we subdivide the input mesh at several steps to avoid asking the solver for a solution which cannot exist. In particular, we subdivide the mesh to ensure that:

- Every tet is adjacent to at most one boundary element.
- No tet has all of its vertices on the boundary.

Note the second condition can be met by tets which are not adjacent to any boundary elements.

Additionally, after the frame field solve and prior to running cube cover, we further subdivide the mesh to ensure that no tet contains more than one singular edge to avoid degeneracies in the parameter domain.

**Choice of Hyperparameters** We record the hyperparameters used in our experiments in Section 5.2. In particular, we set the weights of the terms of our “as-octa-as-possible” fairness term to be small relative to the smoothness weight. We find this to be a good choice

for recovering smooth fields with small constraint residual, but if these weights are set too small we find that frames may begin to degenerate at convergence.

There is a sense in which there is “no free lunch” when it comes to designing volumetric frame field objectives, in the sense than many reasonable seeming desirata are not mutually compatible and must be relaxed relative to each other in order to ensure problem feasibility. In particular:

- The lack of flexibility in defining conformal maps in volumes suggests that we should not expect there to exist smooth, boundary aligned, solutions that are exactly octahedral. For our tet based discretization, it can be shown that the space of exactly integrable orthogonal frame fields is not smooth.
- As illustrated in the case of line fields in Fig. 17, the norms of integrable fields must go to zero in the neighborhood of a singular curve, and so we should not expect to recover solutions which have unit norm everywhere.
- The plane constraint on the boundary might not be possible to satisfy everywhere (see for instance the parameterization show in the inset in section 5.2).

To summarize, in general, we find that a good starting point for choosing hyperparameters is to set  $\lambda_{\text{smooth}} > \lambda_{\text{orth}} > \lambda_{\text{unit}} \sim \lambda_{\text{plane}}$  (where we typically assume WLOG  $\lambda_{\text{smooth}} = 1$ ), and tuning may be required if the default settings do not produce acceptable fields. In Figure 19 we illustrate an example of a model where adjusting parameters can reduce the number of artifacts from degenerate frames on a low resolution mesh.

**Robust Newton Solver** The central numerical difficulty that we have yet to address is that when solving the MINT MESH optimization problem the nonlinear nature of the problem may result in an indefinite Hessian matrix.

This issue arises in many applications in computer graphics, and there are broadly two popular approaches in the graphics literature for modifying non-PD Hessians to perform newtons solves. The first involves projecting the Hessian onto the cone of positive semi-definite (PSD) matrices block-wise [TSIF05, CLL\*24]. While PSD projection can improve robustness, projected Newton methods might exhibit poor convergence near the optimum compared to the standard Newton’s method with an unmodified Hessian [LLF\*23].

The second approach is a form of Hessian modification ( $H + reg \cdot I$ ), where a scaled identity matrix ( $reg \cdot I$ ) is added to the Hessian via a line search (LS) [Noc06, MTG\*11]. The regularization term  $reg$  is dynamically adjusted: increased if the system is found to be non-PD or if steps yield insufficient progress, and decreased otherwise. This adaptively interpolates between a full Newton step ( $reg \rightarrow 0$ ) and a gradient descent step ( $reg \rightarrow \infty$ ). However, excessive regularization can significantly slow down progress in the large  $reg$  limit, and may exhibit even worse convergence behavior than the approach of newton with block-wise PSD projection.

Our solver aims for a balance by employing a hybrid approach inspired by Longya et. al. [LLF\*23], tailored to the task of symmetric frame field design. In particular, we take the approach of dynamically switching between using the PSD-projected Hessian

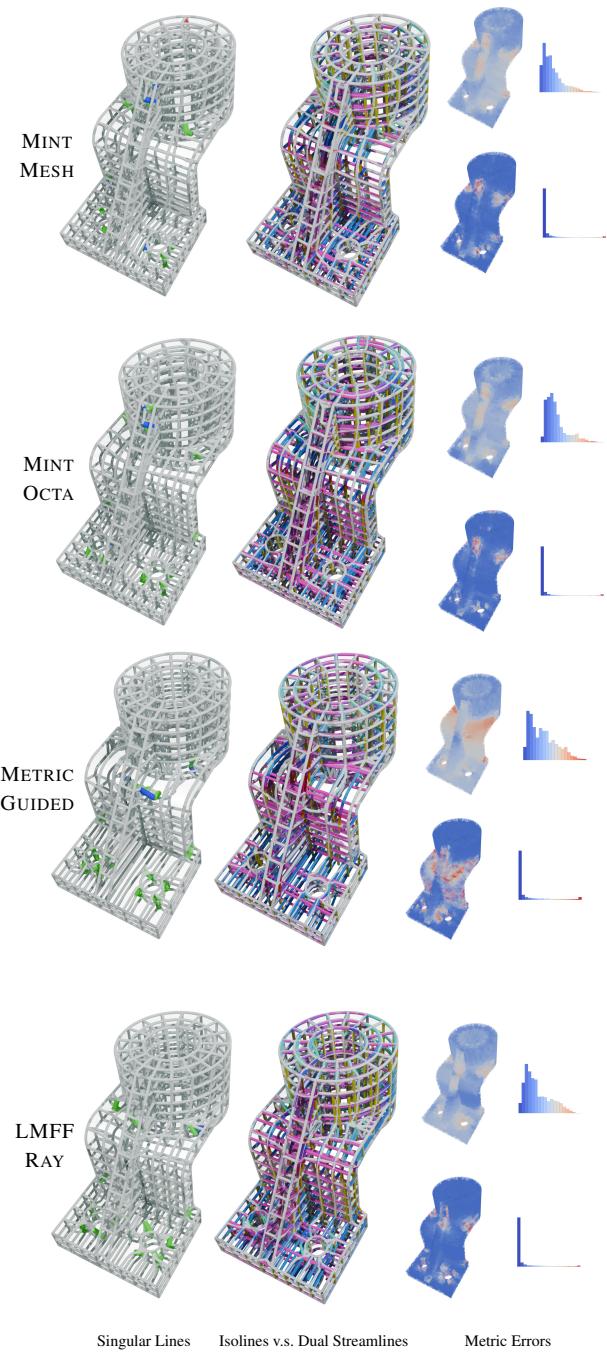
and a LS-projected, diagonally-regularized, Hessian. Specifically, we take the following approach for implementing the newton step of our penalty method:

- Initialize with PSD projection.
- When using the PSD projection, a small fixed diagonal perturbation ( $\text{proj\_reg} \cdot I$ , e.g.,  $10^{-12}I$ ) is added for numerical stability before attempting the linear solve.
- When using the LS approach, the Hessian is regularized by adding  $\text{reg} \cdot I$ , where  $\text{reg}$  starts at a small initial value ( $10^{-8}$ ) and is dynamically adjusted throughout the penalty method solve (e.g. value persists across newton solves).
- Switching Logic: The solver switches strategies based on performance and numerical stability, and persists between newton steps:
  - If the Cholesky factorization fails (matrix not PD): If using PSD projection, switch to LS mode and increases  $\text{reg}$ . If already in LS mode, increase  $\text{reg}$ . If  $\text{reg}$  becomes excessively large (e.g.,  $> 10^8$ ) in LS mode, check during each step if PSD projected hessian has smaller  $L_\infty$  norm w.r.t. unprojected hessian than LS projected hessian. If so switch back to using PSD proj hessian, otherwise keep increasing  $\text{reg}$ . If  $\text{reg} > 10^{30}$ , exit.
  - If the backtracking line search fails to find a step ( $\alpha = 0$ ), this indicates the computed direction is not a descent direction, and the solver toggles between PSD projection and LS mode.
  - If progress is slow while using PSD projection (e.g., small gradient norm ( $10^{-3}$ ), small relative energy decrease ( $10^{-5}$ ), or very small line search step size  $\alpha < 10^{-5}$ ), the solver switches to LS mode, as a heuristic to improve convergence.

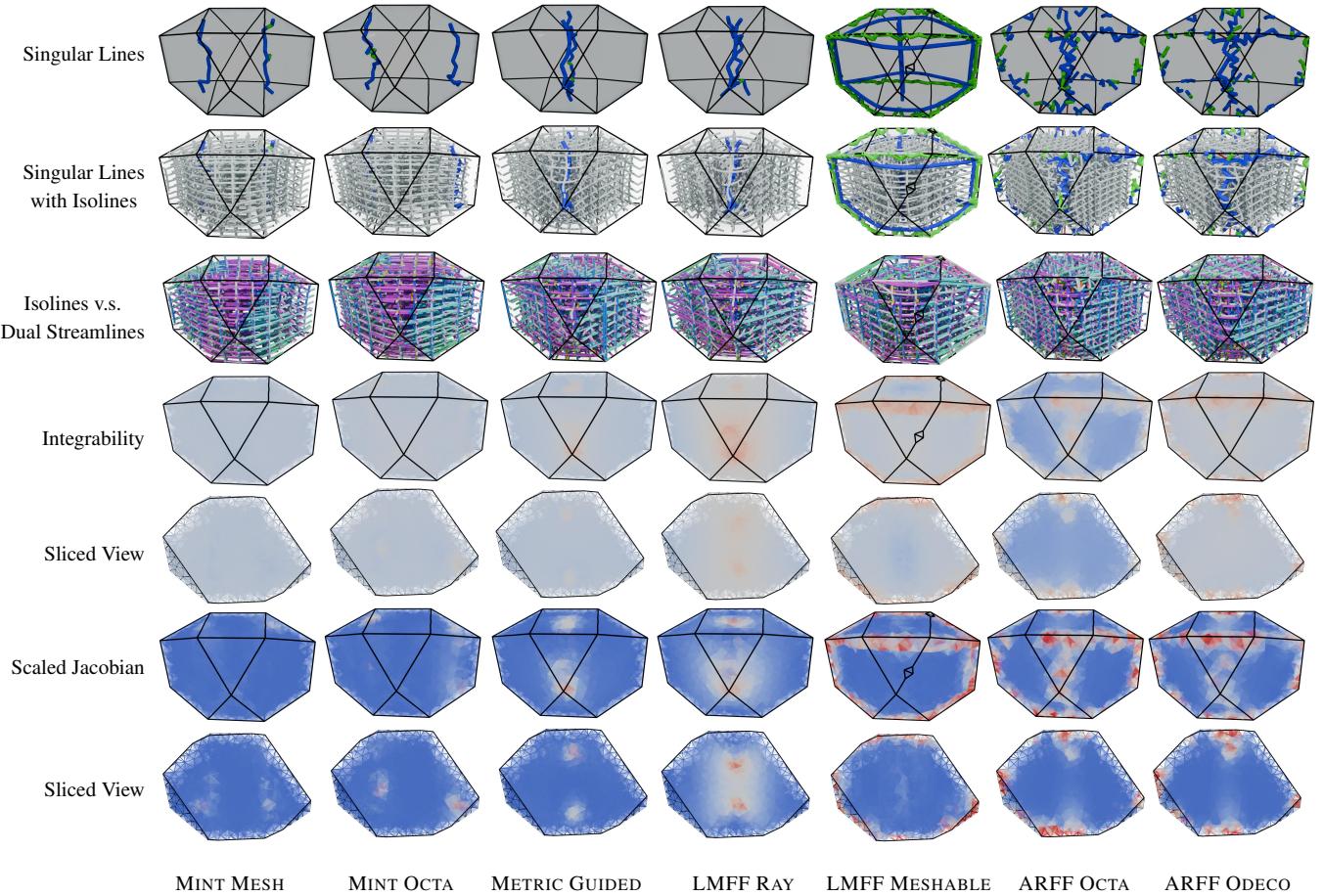
- Regularization Adjustment (LS mode): On successful steps in LS mode, the regularization  $\text{reg}$  is decreased (e.g., halved), moving towards a standard Newton step (while  $\text{reg}$  is larger than a minimum value,  $\text{reg}_{\min}$ , e.g.,  $10^{-16}$ ). If the energy stalls during convergence checks (e.g. exit newton step due to function progress being below the threshold),  $\text{reg}$  is reset to  $\text{reg}_{\min}$  to ensure that  $\text{reg}$  is set as small as possible between newton solves to improve convergence rate.

This hybrid projection strategy attempts to leverage the robustness of both methods while aiming for fast convergence, particularly near the solution where the unmodified Hessian is more likely to be positive definite and  $\text{reg}$  can be driven towards its minimum value. The backtracking line search ensures sufficient decrease according to the Armijo condition.

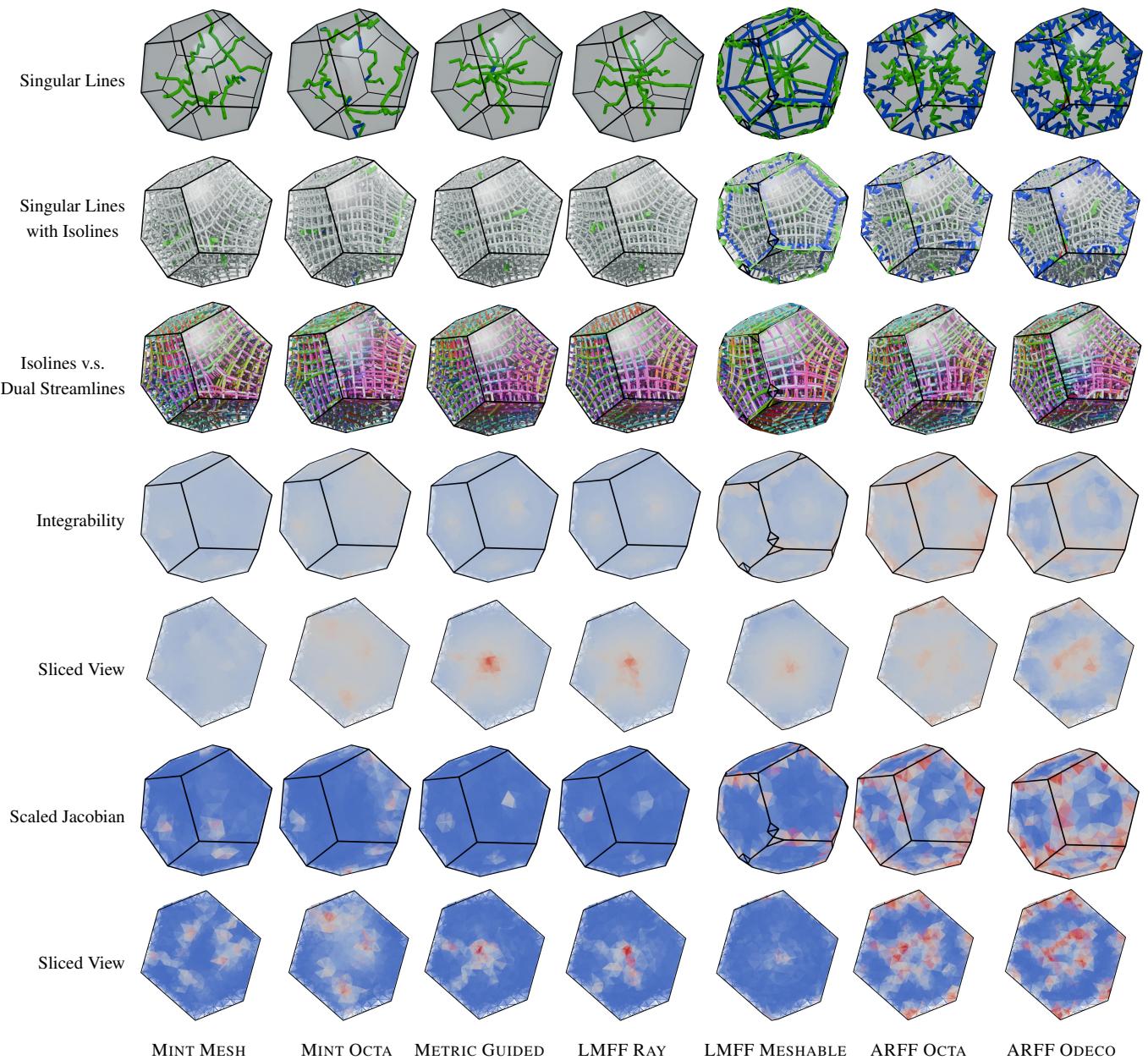
Note that this algorithm only uses the PSD projection as an acceleration strategy and ultimately will switch to using LS with a sufficiently large regularizer if the PSD projection starts converging slowly. As such this can be thought of as an accelerated version of the standard approach of adding a diagonal regularizer to make a hessian PD (e.g. from section 3.4 in [Noc06]). We default to taking 100 newton steps per inner iteration, but as an optimization, we only take 25 at the early and late newton solves. On the other hand, to improve convergence, we employ quite conservative exit criteria (e.g.  $L^2$  norm of gradient  $< 10^{-8}$ , function progress during a step is smaller than  $10^{-14}$ ), and only increase the penalty weight by  $\sqrt{2}$  between newton solves.



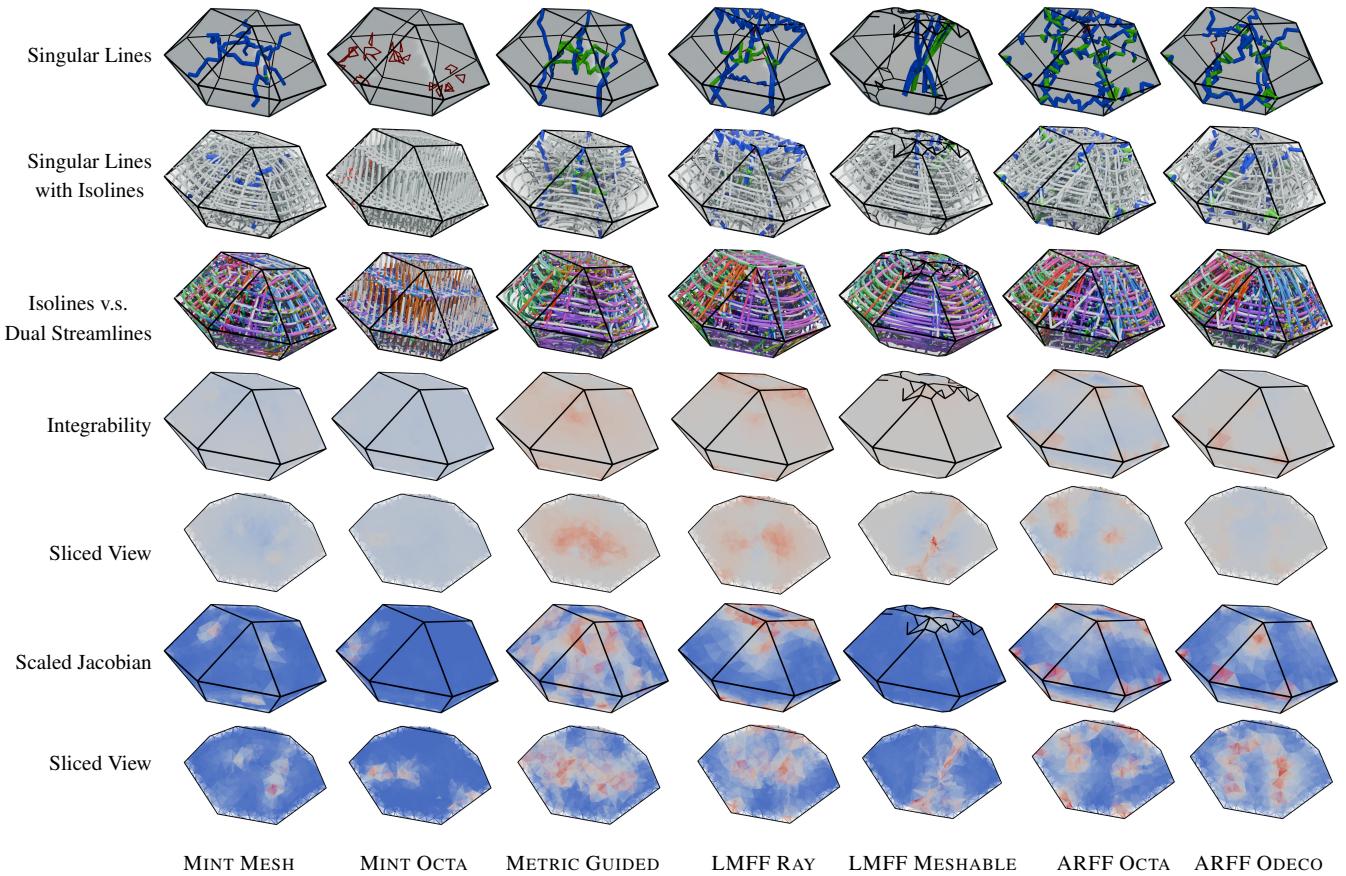
**Figure 12:** Optimal frame fields on the i20 model from the HexMe dataset computed using different methods. Left: Singular lines are shown with isolines to better illustrate the singular structures. Middle: A visual comparison between parameterized isolines and the dual streamlines of the optimal fields. Right: Error plots for different metrics: top—integrability error ( $Q_{\text{int}}$ ), bottom—scaled Jacobian ( $Q_{\text{jac}}$ ).



**Figure 13:** Results on the bilunabirotunda model using different methods. The black lines indicate the silhouette of the output mesh. Among all the methods, MINT MESH achieves the best results, exhibiting smaller integrability and scaled Jacobian errors. In contrast, LMFF LOCAL STEP alters the mesh structure, resulting in a different mesh silhouette. You can see that our methods both find qualitatively different singular structures relative to baselines, with the MINT MESH solution being more symmetric than MINT OCTA.



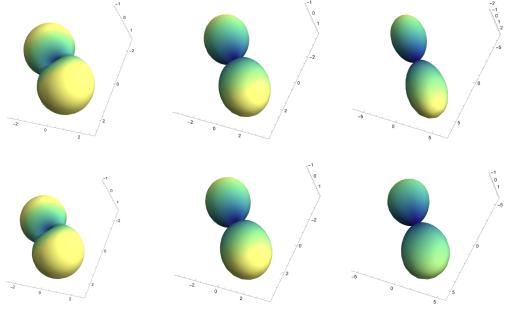
**Figure 14:** Results on a Dodecahedron model. Once again, MINT MESH achieves the best results, exhibiting smaller integrability and scaled Jacobian errors with qualitatively different singular structures relative to baselines, with the MINT MESH solution being more symmetric and reflecting the geometry of the input shape more faithfully than MINT OCTA. Unlike MINT MESH, the other methods produce solutions which concentrate a large amount of integration error near the center of the mesh.



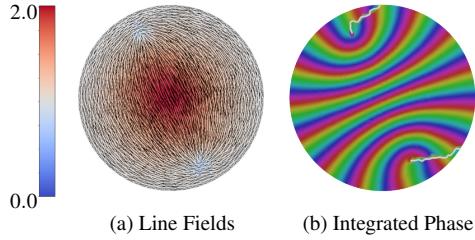
**Figure 15:** Results on the square-gyrobicupola model using different methods. In this case, although MINT OCTA achieves the lowest integrability error, its frame fields are significantly misaligned with the boundary, and the solution it finds has numerous degenerate frames (the red curves in the top figure often appear when frame degenerate). Meanwhile, LMFF LOCAL STEP alters the mesh structure, resulting in a different and noisier mesh silhouette. MINT MESH produces a parameterization (in the second row) which is considerably more regular than the alternative approaches.

**Table 1:** We compare the quality of parameterizations produced by several octahedral symmetric frame field design baselines on the polyhedral dataset. We evaluate the quality of parameterizations with and without an integer grid constraint, e.g. the top row for each method reports results to solving for a map whose gradient is as close as possible to the field produced by the given method, as in Equation 14, and the bottom row reports the same but now when we additionally enforce integer constraints on the boundary and at singularities, as is typically done in hex meshing applications. We highlight the method with the best average performance in each column respect to *seamless* and *intgrid*. Entries record mean of quality metric over the dataset, and numbers in parentheses are the variances. Note that the fraction of inverted elements is already an aggregate statistic so we don't report average variance for that. As shown in the table, we achieved best results over all the other methods except for the anisotropy of non constrained parametrization. We also find that on average enforcing integrability results in parameterizations that are less smooth, but better along the other quality metrics.

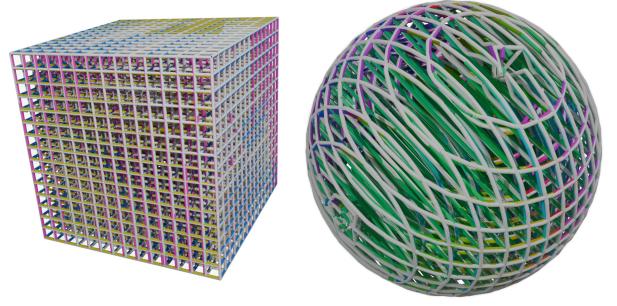
Method	Int Err	SJac	Aniso	DetDev	Inv elem	Smooth	BAlign
MINT MESH	<u>0.172 (0.012)</u>	<u>0.082 (0.023)</u>	0.374 (0.022)	<u>0.275 (0.069)</u>	<u>0.012</u>	0.266 (0.049)	<u>0.029 (0.004)</u>
	<b>0.318 (0.012)</b>	<b>0.174 (0.047)</b>	0.513 (0.028)	<b>0.434 (0.331)</b>	0.098	0.295 (0.072)	<b>0.013 (0.008)</b>
MINT OCTA	0.224 (0.012)	0.100 (0.026)	0.357 (0.023)	0.282 (0.251)	0.026	<u>0.242 (0.050)</u>	0.050 (0.007)
	0.381 (0.013)	0.210 (0.054)	0.551 (0.027)	0.497 (0.805)	0.146	<b>0.268 (0.069)</b>	0.017 (0.010)
METRIC GUIDED	0.243 (0.016)	0.109 (0.032)	<u>0.294 (0.024)</u>	0.305 (0.103)	0.022	0.289 (0.070)	0.043 (0.006)
	0.415 (0.015)	0.237 (0.062)	0.574 (0.029)	0.629 (1.068)	0.162	0.300 (0.080)	0.026 (0.014)
LMFF RAY	0.265 (0.016)	0.123 (0.034)	0.320 (0.026)	0.324 (0.108)	0.023	0.295 (0.069)	0.047 (0.008)
	0.455 (0.014)	0.253 (0.062)	0.593 (0.031)	0.664 (42.290)	0.171	0.287 (0.069)	0.028 (0.015)
LMFF MESHABLE	2.419 (12067.3)	0.182 (0.052)	0.418 (0.030)	0.933 (121.844)	0.073	23.917 (615456.77)	0.062 (0.015)
	3.011 (31139.68)	0.205 (0.052)	<b>0.482 (0.035)</b>	0.990 (89.91)	<b>0.095</b>	26.292 (1318836)	0.024 (0.013)
ARFF OCTA	0.326 (0.029)	0.182 (0.082)	0.299 (0.048)	0.380 (0.098)	0.062	0.373 (0.124)	0.122 (0.029)
	0.447 (0.027)	0.299 (0.098)	0.579 (0.049)	0.612 (0.403)	0.153	0.373 (0.115)	0.088 (0.045)
ARFF ODECO	0.268 (0.028)	0.178 (0.080)	0.349 (0.042)	0.376 (0.106)	0.059	0.335 (0.093)	0.105 (0.025)
	0.431 (0.022)	0.303 (0.097)	0.561 (0.045)	0.573 (0.389)	0.160	0.336 (0.086)	0.077 (0.042)



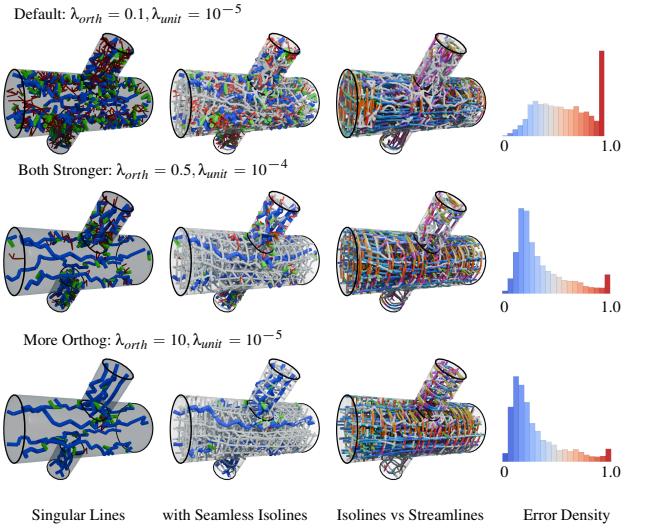
**Figure 16:** Here we illustrate  $[L_2, L_4, L_6]$  of  $\mathbf{F}$  and  $\mathbf{G}$  from Eq. 25 similar to the technique used in the frame field literature, e.g. the counter example described above by taking each point in  $S^2$ , and contracting it with the lifted tensors until it returns a scalar, and rescaling the input vector by that scalar. Unexpectedly for these two frames  $L_2(\mathbf{F}) = L_2(\mathbf{G})$  and  $L_4(\mathbf{F}) = L_4(\mathbf{G})$  but  $L_6(\mathbf{F}) \neq L_6(\mathbf{G})$  plotting the moments as functions over sphere where the color corresponds to the scalar value that arises from contracting the given tensor with the point on the surface of the sphere an appropriate number of times.



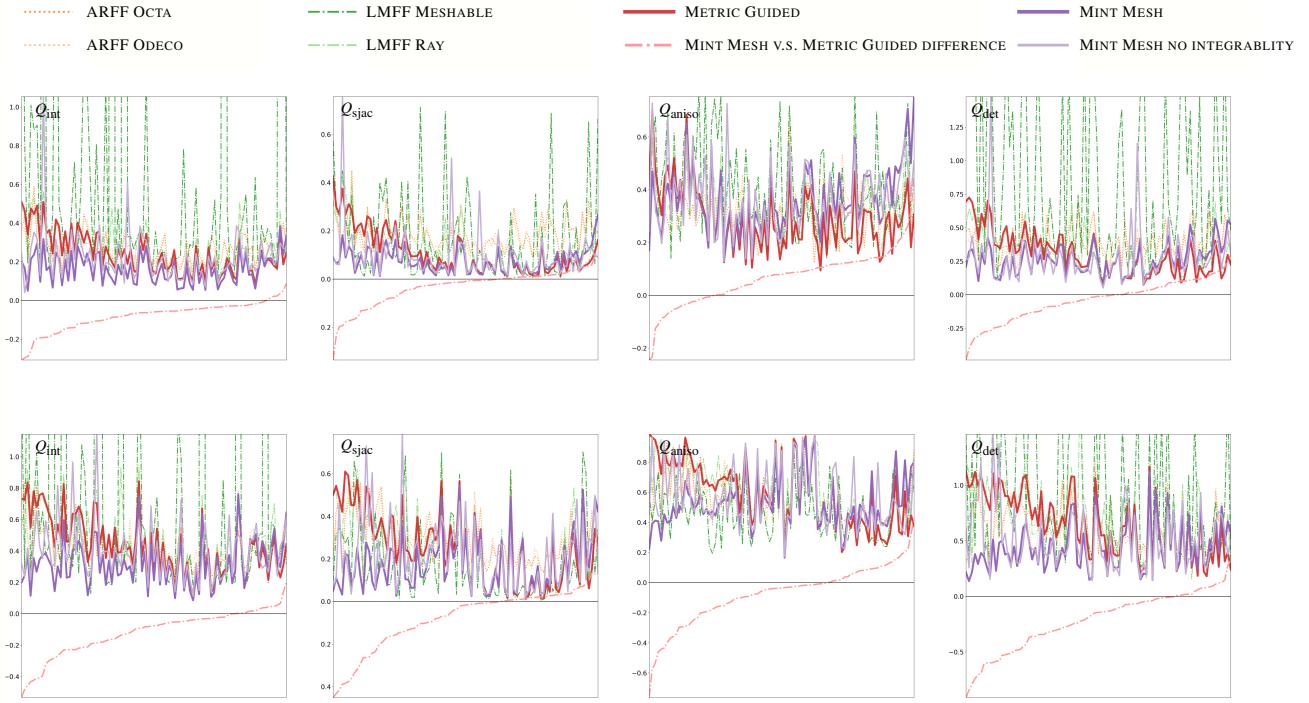
**Figure 17:** In the smooth setting, there is no way to assign vector field on the boundary of a circle, so that this field integrates to a nonzero multiple of  $2\pi$ , and extends smoothly to an integrable (over  $S^1$ ) function on the interior. Somewhat surprisingly, the same is not true in the case of integrable line fields. Here we show an integrable line field which conforms to the prescribed circulating boundary conditions whose gradient norms go to zero in the neighborhoods of two index  $+\frac{1}{2}$  singularities (left). If we integrate up over  $S^1$ , we get the result (right), plotting phase as color on one of the covers, and indicating the branch cuts.



**Figure 18:** We show parameterizations that result from minimizing only  $E_{smooth}$  on two simple examples from random initialization with unit normal boundary conditions.



**Figure 19:** In this figure we probe the sensitivity of MINT MESH to choice of weights in the objective, on the *s08o\_cross\_cyls\_dr* model from the *HexMes* dataset. In the (top row), we document results from our default parameter setting. In the (middle row), we increase both of the parameters slightly, and see significant improvement, but some artifacts remain. In the (bottom row), increase the orthogonality weight by 100x relative to our default setting. While these both improve the results on this example, particularly the latter parameter choice may converge to less smooth solutions on average on other examples in our test set. Field quality can be improved dramatically by adjusting the model hyperparameters to promote mesh orthogonality on this example, and there remains research to be done in exploring robust formulations of symmetric frame field design objectives.



**Figure 20:** Effect of integrability on parameterization quality. Plots from ablation study on 120 model polyhedral dataset comparing a number of parameterization quality metrics of the parameterizations which result from frame field design. Top row plots show per model average of quality measures with respect to seamless parameterization, bottom row is with respect to integer-grid parameterization.