



Analytic methods for geometric modeling via spherical decomposition[☆]



Morad Behandish^{*}, Horea T. Ilieş

Computational Design Laboratory (CDL), University of Connecticut, Storrs, CT 06269, USA

HIGHLIGHTS

- We propose a grid-free discretization scheme for analytic geometric modeling.
- Solids are approximated with countable unions of 3D balls cut from 4D cones.
- The unions turn into 3D slices of 4D Minkowski sums of knots and a template cone.
- The Minkowski formulation embeds well into cross-correlations between solids.
- The analytic formulation follows using convolution algebra and Fourier Transform.

ARTICLE INFO

Keywords:

Analytic methods
Shape correlation
Spherical sampling
Fourier transform
Collision detection

ABSTRACT

Analytic methods are emerging in solid and configuration modeling, while providing new insights into a variety of shape and motion related problems by exploiting tools from group morphology, convolution algebras, and harmonic analysis. However, most convolution-based methods have used uniform grid-based sampling to take advantage of the fast Fourier transform (FFT) algorithm. We propose a new paradigm for more efficient computation of analytic correlations that relies on a grid-free discretization of arbitrary shapes as countable unions of balls, in turn described as sublevel sets of summations of smooth radial kernels at adaptively sampled 'knots'. Using a simple geometric lifting trick, we interpret this combination as a convolution of an impulsive skeletal density and primitive kernels with conical support, which faithfully embeds into the convolution formulation of interactions across different objects. Our approach enables fusion of search-efficient combinatorial data structures prevalent in time-critical collision and proximity queries with analytic methods popular in path planning and protein docking, and outperforms uniform grid-based FFT methods by leveraging nonequispaced FFTs. We provide example applications in formulating holonomic collision constraints, shape complementarity metrics, and morphological operations, unified within a single analytic framework.

© 2015 Elsevier Ltd. All rights reserved.

1. Introduction

Analytic modeling relies on describing shape and configuration pointsets as sublevel sets of functions and formulating fundamental operations (e.g., pertaining to detecting collisions, similarity, complementarity, or symmetry) in terms of correlations between those functions. For example, Minkowski operations [1] that are central to mathematical morphology are formalized as convolutions of constituent functions and computed efficiently

in the Fourier domain [2]. Minkowski operations have been used extensively to formulate important problems in robot path planning [3], mechanism workspace design [4], virtual reality (graphics/haptics) [5,6], protein docking [7], packaging and nesting [8], and more. Unfortunately, their combinatorial computation even for 3D polyhedral objects quickly becomes impractical with increasing number of polygons [2]. This can be alleviated using classical FFTs [9] for numerical implementation of convolutions, which take advantage of uniform spatial sampling. However, this and several related correlation-based problems can be solved more efficiently by using a spherical decomposition of the shape and nonuniform FFTs [10]. Here we briefly review the roots of the main ideas, with a focus on collision detection (CD) and shape complementarity (SC) literature.

[☆] This paper has been recommended for acceptance by Scott Schaefer and Charlie C.L. Wang.

^{*} Corresponding author.

E-mail address: morad@engineer.uconn.edu (M. Behandish).

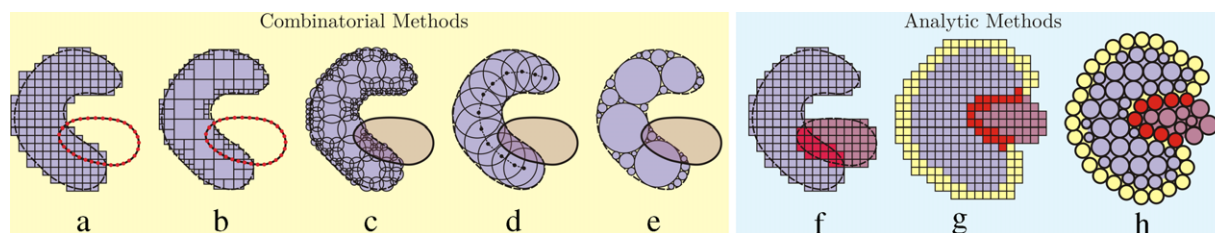


Fig. 1. Research in combinatorial methods has shown that CD tests over uniform grids (i.e., ‘voxmaps’) [11] (a) and octrees (special case of OBB-trees) [12] (b) can be made more efficient if voxels are replaced with spherical primitives, e.g., built around octrees [13] (c), sampled over the MA [14,15] (d) or packed inside using distance fields [16] (e). However, the more nascent analytic methods are still mostly reliant on uniform grids for, e.g., CD testing for solids by integrating the intersection [17] (f), and SC scoring for proteins by integrating skin overlaps [18] (g). The latter has been outperformed by grid-free correlations of atoms grouped with equal radii [7] (h). We show that constructions in (c)–(e) with arbitrary radii can also be interpreted analytically as a convolution and solved by nonuniform FFTs after a geometric lifting.

1.1. Background

At one end of the research spectrum are combinatorial techniques that use surface meshes or higher-order algebraic parametrizations to resolve collisions or identify matching features. Examples are polyhedral CD methods based on Voronoi-clipping/marching [19] and oriented bounding box (OBB) trees [12], or spatial enumeration-based techniques such as the Voxmap PointShell (VPS) [11]. VPS works by a pairwise test between a shell of vertices for the moving object against a map of voxels that discretizes the stationary obstacle, and is popular in physically-based modeling in virtual environments [20]. Others have identified more efficient techniques for time-critical CD by using hierarchical bounding spheres sampled on octrees [13] or on the medial axes (MA) [14,15], and interior sphere packing guided by distance fields [16] (Fig. 1(a)–(e)). These ‘sphere-tree’ based methods have been shown to outperform voxmap or OBB-tree based techniques in real-time applications [16] – particularly because primitive collision predicates are simplified to center-distance tests as a result of the orientational symmetry of balls – and are considered state-of-the-art in practice. For a more complete survey on CD methods, we refer the reader to [21].

On the other end of the spectrum are analytic methods that have been more popular in robotics [3]. Unlike the combinatorial approach that searches for a collision certificate point (or lack thereof) in the intersection of the objects, the analytic approach treats the collision predicate as a Boolean combination of inequalities over some configuration space of the objects [22]. The obstacle avoidance in path planning is, for example, treated as an optimization problem subjected to holonomic collision constraints formulated analytically as a convolution of the robot and its workspace [17]. Most convolution-based methods have so far focused on generating uniformly sampled configuration bitmaps for all spatial positions and orientations simultaneously, which can be cumulatively computed with asymptotically optimal FFTs [9]. However, a complete description of the configuration obstacle is overkill for real-time CD. A recent work [22], also reliant on uniform grid-based sampling, formally reframes the approach for time-critical CD (Fig. 1(f)), but has not yet been compared with sphere-tree methods, nor applied to real-time applications.

In an independent line of research, numerous analytic methods for molecular surface analysis and SC-based protein docking have been developed, whose outcomes are platforms that use grid-based occupancy enumeration and leverage classical FFTs [9] such as ZDock [18], or more recent grid-free techniques that rely on nonuniform FFTs [10], such as F²Dock [7] (Fig. 1(g), (h)). The latter exploits the spherical shape of the atomic building blocks and implicitly represents the proteins as summations of radial kernels centered around atoms, assigning different weights to core and skin atoms. The SC score is obtained by cross-correlating these functions from different proteins, which turns into a convolution discretized over the center points. It has been shown that grid-free

methods outperform uniform grid-based methods [7] by taking advantage of the spherical geometry. A comprehensive survey on advances in protein docking is available in [23].

Although objects of arbitrary shape, unlike molecules, cannot be represented *exactly* as finite unions of balls, the sphere-tree methods for time-critical CD were shown to be more successful in progressively approximating the shape, when compared to uniform grid- or octree-based voxelization, with a faster convergence and a better use of computational resources [14]. Motivated by this observation, we present a generic framework for representing arbitrary shapes with finite (or countably infinite, in the limit) radial kernels, formulated as a convolution of a discrete pointset and the primitive kernel in a higher-dimensional space. The latter is described as a *geometric lifting* trick in Section 2.2, and is deemed necessary due to the inevitable size difference between primitive balls, unlike the simpler case for the proteins. We show that this approach offers ‘the best of both worlds’ by combining the computational efficiency of the sphere-tree techniques for time-critical applications (i.e., with a single configuration query) with that of the analytic methods for cumulative configuration space constructions (i.e., requiring a complete map for all spatial relations), unified under a single paradigm with analytic formalism that applies to a multitude of applications.

1.2. Contributions

The main contributions of this paper are to (1) present an analytic shape correlation paradigm centered around a nonuniform discretization scheme¹ that relies on progressive spherical approximations with balls of different sizes; and (2) a uniform and efficient approach to solving a variety of problems that deal with detecting collisions, shape similarity/complementarity, and shape morphing, examples of which we describe in Section 4.

Moreover, we show that the spherical discretization offers an algebraic structure that is closed under Minkowski sum/product operations, and at the same time offers more appealing properties than uniform grid- or octree-based discretizations. As the continuous geometry (of both shapes and configurations) is abstracted away by the balls, the computational implementation solely relies on convolution algebra over discrete sets specified completely by ball center coordinates and radii, allowing the use of the efficient nonequispaced FFT (NFFT) algorithm [10] on the highly parallel graphics processing units (GPU) [24].

¹ What we mean by a ‘discretization scheme’ is not a particular decomposition algorithm or approximation method, but a generic formalism for reconciling such a nonuniform discretization (in contrast to the extensively used uniform sampling) to analytic modeling, using Minkowski sums and convolutions. We do present one new algorithm in Appendix A; nevertheless, other methods [13–16] are also applicable under the same scheme.

Unlike combinatorial methods whose complexities typically depend on the syntactic size of the representation (e.g., polygon count) fixed upfront, our method allows for a choice of complexity in real-time based on the affordable resources, by proceeding deep enough down the sphere-tree—which can be constructed using any algorithm of choice, such as [13–16] or our own presented in Appendix A. On the other hand, unlike grid-based analytic methods whose arithmetic complexities scale with object size and grid resolution, our method enables filling large regions with large balls and efficiently allocating more primitives to capture features of smaller size with higher fidelity.

Finally, by working in the Fourier domain, aside from converting convolutions to simple pointwise products and differentiations to simple multipliers, the method allows for ‘graceful’ degradation of the accuracy by truncating the frequency domain representations, enabling another trade-off mechanism between running time and precision.

1.3. Outline

In Section 2 we present the formalism for discretization of an arbitrary shape as a countable union of balls, its interpretation as a Minkowski sum, and its analytic description as a convolution. In Section 3 we formulate correlation predicates in terms of Minkowski sums across multiple shapes relatively positioned and oriented in arbitrary ‘poses’ together with their convolution forms, and use the results from the previous section to carry the discretization into the configuration space. In Section 4 we provide examples of applying these tools to solving fundamental problems, whose efficient GPU implementations are shown to outperform some of the state-of-the-art methods in those areas as demonstrated in Section 5. A more detailed outline of the subsections is provided at the beginning of each section.

2. Shape discretization

In this section, we present what we mean by ‘analytic modeling’ in 2.1. We define a discretization scheme in 2.2 as a 3D slice of a 4D Minkowski sum of a countable set of ‘knots’² and a primitive cone that represents balls of all permissible sizes stacked along the 4th dimension. In 2.3, we briefly introduce the sampling approaches that generate such discretizations from arbitrary representations. We incorporate motions in 2.4 and show the advantages of spherical symmetry in the presence of rotations. In 2.5, we present the Fourier expansion of the spherical sampling that facilitates development of fast algorithms.

2.1. Analytic solid modeling

As usual, we assume solids to be ‘r-sets’, defined by Requicha [26] as compact (i.e., bounded and closed) regular semianalytic subsets of the Euclidean 3-space $\mathcal{S} \subset \mathcal{P}(\mathbb{R}^3)$.³ The regularity condition (i.e., $S = rS$) ensures *continuous homogeneity*,⁴ while the semianalytic requirement guarantees *finite describability* of the set [26], as well as its medial axis (MA) and medial axis transform (MAT) [28]. The latter is defined as an embedding of the MA in the

² The terminology is borrowed from Potts et al. [25], where convolution with radial kernels at ‘nonequispaced knots’ (which underlies our development) is carried out using the nonequispaced FFT [10].

³ The collection $\mathcal{P}(A) = \{B \mid B \subset A\}$ denotes the ‘power set’ of a set A , i.e., the set of all subsets of A .

⁴ $rS = \kappa rS$ denotes the ‘regularized’ (i.e., closure of interior of) S . The ‘boundary’ $\partial S = \kappa S \cap \kappa cS$ is homogeneously 2D for a 3D r -set, separating the open ‘interior’ iS , from the open ‘exterior’ cS [27].

4D space with the radius of the maximal ball conceptualized as a new coordinate, and contains enough information to reconstruct the solid S , hence can be used to develop a discretization/sampling scheme in Section 2.2.

To enable an *analytic* formulation of the geometric modeling operators in Section 3, a solid $S \in \mathcal{S}$ can be implicitly described as a sublevel set of a real-valued function $f_S : \mathbb{R}^3 \rightarrow \mathbb{R}$ (called the ‘defining function’ of S), first introduced by Comba [29] for CD between convex objects, and used later by Ricci [30] to implement Boolean combinations. By restricting to compactly supported non-negative functions $f_S : \mathbb{R}^3 \rightarrow [0, \infty)$, and using regularized 0-sublevels, i.e., $S = U_0^*(f_S) = rU_0(f_S)$, where

$$U_t(f_S) := \{\mathbf{x} \in \text{domain of } f_S \mid f_S(\mathbf{x}) > t\}, \quad (1)$$

the regularized set operations are supported by the analytic expressions

$$S_1 \cup^* S_2 = U_0^*(f_{S_1} + f_{S_2}), \quad \text{i.e., } f_{S_1 \cup^* S_2} = f_{S_1} + f_{S_2}, \quad (2)$$

$$S_1 \cap^* S_2 = U_0^*(f_{S_1} f_{S_2}), \quad \text{i.e., } f_{S_1 \cap^* S_2} = f_{S_1} f_{S_2}. \quad (3)$$

A popular choice for the defining function is the ‘indicator function’ $f_S := \mathbf{1}_S : \mathbb{R}^3 \rightarrow \{0, 1\}$, where $\mathbf{1}_S(\mathbf{x}) = 1$ if $\mathbf{x} \in S$ and $\mathbf{1}_S(\mathbf{x}) = 0$ otherwise. However, the discontinuity of the indicator function makes it difficult to compute the gradients of correlation functions between solids. Lysenko [22] circumvented this problem in his collision constraint formulation by employing ‘bump functions’ $f_S \in C_0^\infty(\mathbb{R}^3)$ instead, which are compactly supported nonnegative functions that are also smooth, offering more appealing differential properties than the discontinuous indicator functions [17], the C^0 – (but not C^1 –) distance fields [30], and the rather complex R-functions [31]. Fig. 2 illustrates indicator and bump functions, and their discretizations introduced in Section 2.2.

A useful result that underlies the collision predicate formulation in Section 4.1 is the *null-volume lemma* [22], which relies on the regularity (hence 3D homogeneity):

Lemma 1. For every r -set $S = U_0^*(f_S)$ with a nonnegative defining function $f_S : \mathbb{R}^3 \rightarrow [0, \infty)$,

$$S \neq \emptyset \Leftrightarrow v(f_S) := \int_{\mathbb{R}^3} f_S(\mathbf{x}) \, d\mathbf{x} > 0. \quad (4)$$

In particular, if $f_S := \mathbf{1}_S$ then $v(f_S)$ exactly gives the volume of S . Letting $S := S_1 \cap^* S_2$ in (4) and applying (3), i.e., $f_S = f_{S_1} f_{S_2}$, yields the collision predicate in terms of the inner product of functions

$$S_1 \cap^* S_2 \neq \emptyset \Leftrightarrow v(f_S) = \langle f_{S_1}, f_{S_2} \rangle > 0, \quad (5)$$

which underlies generalized correlation predicates for a range of applications discussed in Section 4.

It is rarely the case in practical applications with arbitrarily complex shapes to have the defining function in closed form. However, one can always decompose the solid into a finite number of simpler primitives, as an immediate consequence of its finite describability [26], and apply (2) to combine the primitive defining functions—to which we refer as ‘discretization’. However, obtaining exact discretizations (e.g., curvilinear cell decompositions) from the popular constructive solid geometry (CSG) or boundary representation (B-rep) schemes [32] is not trivial.⁵ An alternative is to use approximate discretizations (e.g., spatial enumerations via uniform grids or octrees) which converge to the r -set in the limit. Next, we introduce a more general discretization scheme that subsumes these enumeration methods with non-intersecting cubic primitives (i.e., voxelization) as special cases, and enables other types of (possibly intersecting) primitives.

⁵ An r -set $S \in \mathcal{S}$ is a ‘tame’ embedding of a polyhedron $\Delta = \bigcup_{1 \leq i \leq n} \Delta_i$ under a homeomorphism $\gamma : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ [26], i.e., $S = \gamma(\Delta) = \bigcup_{1 \leq i \leq n} \gamma(\Delta_i)$, and $f_S(\mathbf{x}) = \sum_{1 \leq i \leq n} (f_{\Delta_i} \circ \gamma^{-1})(\mathbf{x})$ due to (2). Although coming up with a closed form for f_{Δ_i} for a tetrahedron Δ_i is trivial, finding γ from CSG or B-rep is not.

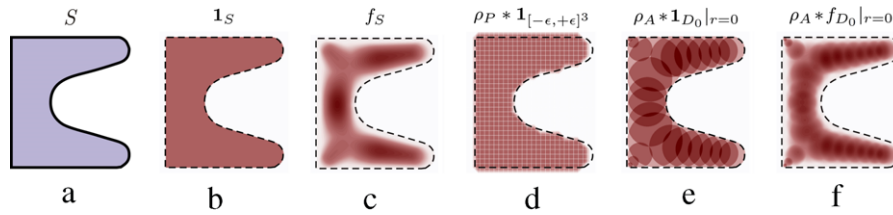


Fig. 2. An r-set (a), its indicator function (b) and bump function (c), discretized using grid-based bitmap sampling (d), and grid-free spherical sampling corresponding to $\alpha \rightarrow \infty$ (e) and $\alpha := 2$ (f) in (10).

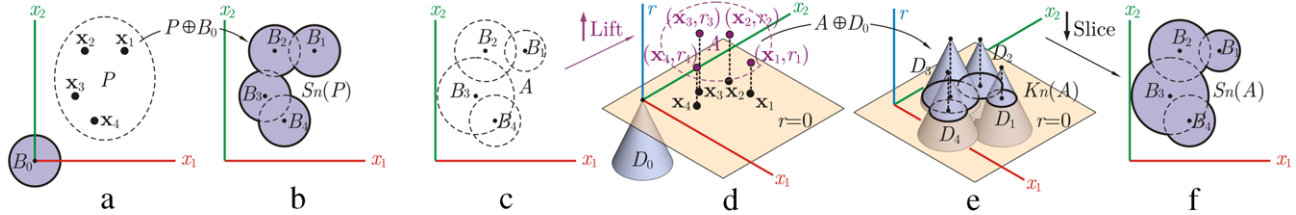


Fig. 3. An equiradius sampling of a 3D r-set is a 3D Minkowski sum $S_n(P) = P \oplus B_0 = U_0^*(\rho_P * f_{B_0})$ (a), (b), while its nonequiradius sampling is a 3D slice through a 4D Minkowski sum $K_n(A) = A \oplus D_0 = U_0^*(\rho_A * f_{D_0})$ of lifted primitives (c)–(f), illustrated here for 2D r-sets.

2.2. Discrete constructions

Consider the case when an r-set can be decomposed as a finite union $S = \bigcup_{1 \leq i \leq n} B_i$ hence $f_S(\mathbf{x}) = \sum_{1 \leq i \leq n} f_{B_i}(\mathbf{x})$ due to (2). The following two cases are of prime significance for our purposes:

Equiradius decomposition. First, let B_i ($1 \leq i \leq n$) be relocated instances of the same base shape $B_0 \in \mathcal{S}$ instantiated by the different translations $\mathbf{x} \mapsto \mathbf{x}_i + \mathbf{x}$, i.e., $B_i = \{\mathbf{x}_i + \mathbf{x} \mid \mathbf{x} \in B_0\}$. Each such translation can be abstracted by a 3D point $\mathbf{x}_i \in \mathbb{R}^3$, hence the discrete pointset $P := \{\mathbf{x}_i\}_{1 \leq i \leq n} \subset \mathbb{R}^3$ (of cardinality $|P| = n$) and the base primitive B_0 contain all the information to reconstruct the solid. We can use the notation $S = S_n(P)$, as if $S_n : \mathbb{R}^{3n} \rightarrow \mathcal{P}(\mathbb{R}^3)$ is a mapping from the discrete space to the shape space, illustrated in Fig. 3(a), (b). A crucial observation is that this mapping can be viewed as a Minkowski sum $S_n(P) = P \oplus B_0$.

To construct an analytic model, let $B_0 = U_0^*(f_{B_0})$ hence a defining function for each primitive instance can be obtained as $f_{B_i}(\mathbf{x}) := c_i f_{B_0}(\mathbf{x} - \mathbf{x}_i)$, where $c_i > 0$ are weight coefficients (arbitrarily assigned, for now). The Boolean summation of the balls from (2) takes the form

$$f_{S_n(P)}(\mathbf{x}) = \sum_{1 \leq i \leq n} f_{B_i}(\mathbf{x}) = \sum_{1 \leq i \leq n} c_i f_{B_0}(\mathbf{x} - \mathbf{x}_i), \quad (6)$$

which can be viewed as a discrete convolution. To make it compatible with continuous convolutions to be introduced in Section 3, let us rewrite (6) as an integral

$$f_{S_n(P)}(\mathbf{x}) = \sum_{1 \leq i \leq n} \int_{\mathbb{R}^3} \delta^3(\mathbf{x}' - \mathbf{x}_i) [c_i f_{B_0}(\mathbf{x} - \mathbf{x}')] d\mathbf{x}',$$

in which the integration variable is $\mathbf{x}' = (x'_1, x'_2, x'_3) \in \mathbb{R}^3$, and $d\mathbf{x}' = dx'_1 dx'_2 dx'_3$ is the infinitesimal volume element. $\delta^3(\mathbf{x}') = \delta(x'_1) \delta(x'_2) \delta(x'_3)$ is the 3D Dirac delta function. If we assume a density function of the form

$$\rho_P(\mathbf{x}) := \sum_{1 \leq i \leq n} c_i \delta_{\mathbf{x}_i}(\mathbf{x}) = \sum_{1 \leq i \leq n} c_i \delta^3(\mathbf{x} - \mathbf{x}_i), \quad (7)$$

where $\delta_{\mathbf{x}_0}(\mathbf{x}) = \delta^3(\mathbf{x} - \mathbf{x}_0)$, to represent the discrete pointset P as a collection of spatial impulses of intensities c_i at positions $\mathbf{x} = \mathbf{x}_i$ (i.e., each point carrying a ‘lumped mass’ of c_i), the Minkowski sum can be analytically expressed as a convolution

$$f_{S_n(P)} = \rho_P * f_{B_0}, \quad \text{i.e., } P \oplus B_0 = U_0^*(\rho_P * f_{B_0}). \quad (8)$$

A particularly favorable choice for the primitive shape (for reasons to be explained in Section 2.4) is a closed L^2 -ball, i.e.,

$$B_i := B(\mathbf{x}_i, r) = \{\mathbf{x} \in \mathbb{R}^3 \mid \|\mathbf{x} - \mathbf{x}_i\|_2 \leq r\} \quad (9)$$

are balls of constant radius $r > 0$ centered at $\mathbf{x}_i \in P$. Let $f_{B_0}(\mathbf{x}) := \psi_\alpha(\|\mathbf{x}\|_2/r)$ where the function $\psi_\alpha : \mathbb{R} \rightarrow [0, 1]$ is a generic cut-off kernel (also referred to as the ‘mollifier’ or the ‘bump’) with the closed form

$$\psi_\alpha(x) = \begin{cases} e^{(1-|x|^{-\alpha})^{-1}} & \text{if } |x| < 1, \\ 0 & \text{otherwise,} \end{cases} \quad (10)$$

which can be thought of as a smoothed extension of the discontinuous cut-off function $\mathbf{1}_{(-1,+1)}(x) = \lim_{\alpha \rightarrow \infty} \psi_\alpha(x)$, and the resulting $f_{B_0}(\mathbf{x}) = \psi_\alpha(\|\mathbf{x}\|_p/r)$ is a mollified extension of the binary indicator $\mathbf{1}_{B_0}(\mathbf{x}) = \lim_{\alpha \rightarrow \infty} f_{B_0}(\mathbf{x})$.

The spatial enumeration schemes over uniform grids – ranging from bitmap encoding for path planning [17] to rasterized density functions for protein docking [18], illustrated in Fig. 1(f), (g) – can be viewed as special cases of this scheme with cubic primitives (i.e., L^∞ - instead of L^2 -balls in (9)) with an additional disjointness condition that is unnecessary for our purposes. Grid-free molecular modeling based on Gaussian densities for protein surface reconstruction [33] and protein docking [7] (Fig. 1(h)) are more closely related to the scheme proposed here, as they use spherical primitives.⁶

Generalizing this grid-free discretization to solid objects of arbitrary geometric complexity would enable more efficient use of the computational resources by adaptively approximating the shape, filling large interior regions with fewer primitives, and allocating resources to capture the details of surface features. A simple solution is to use a recursive decomposition (e.g., an octree) and take the leaf cells (or balls enclosed by their bounding spheres) as primitives (Fig. 1(c)), collected into groups of constant radii according to their level in the tree, for (8) to apply. Hubbard [14] showed that octree-based spherical approximation is nonoptimal in terms of Hausdorff convergence, and compares poorly with sampling the centers of primitive balls over the MA (Fig. 1(d)).

⁶ Except that these methods use Gaussian kernels rather than compactly supported cut-off kernels such as the one in (10), and use the 1- (instead of the 0-) isovalue to define the molecular surface.

However, the latter requires a generalization of (8) that supports different sizes for the balls, which we address next.

Nonequiradius decomposition. The generalization is enabled by a simple *geometric lifting* trick. This time, let B_i ($1 \leq i \leq n$) be translated and scaled instances of the base shape $B_0 \in \mathcal{S}$, i.e., instantiated by the affine transformation $\mathbf{x} \mapsto \mathbf{x}_i + (r_i\mathbf{x})$, i.e., $B_i = \{\mathbf{x}_i + (r_i\mathbf{x}) \mid \mathbf{x} \in B_0\}$. Each such transformation can be abstracted by a 4D point $\mathbf{a}_i := (\mathbf{x}_i, r_i) \in \mathbb{R}^4$. The discrete pointset $A := \{\mathbf{a}_i\}_{1 \leq i \leq n} \subset \mathbb{R}^4$ (of cardinality $|A| = n$) contains all the information to reconstruct the solid, and $S = S_n(A)$ can be viewed as a Minkowski product [1] of the form $S_n(A) = A \otimes \gamma_0(B_0)$, defined over the group of the aforementioned instance transformations $G \cong \mathbb{R}^4$, where $\gamma_r : \mathbb{R}^3 \hookrightarrow \mathbb{R}^4$, $\gamma_r(\mathbf{x}) = (\mathbf{x}, r)$, and $\gamma_0(B_0) = B_0 \times \{0\}$ is a trivial embedding in G .

A more helpful way of looking at this formulation is to think of each primitive $B_i \subset \mathbb{R}^3$ as a cross-section (i.e., a 3D ‘slice’ orthogonal to the r -axis at $r = 0$) through a hypothetical hypercone $C_i \subset \mathbb{R}^4$ whose apex is located at $(\mathbf{x}_i, r_i) \in A$, illustrated in Fig. 3(c), (d). To ensure compactness of the 4D objects, let us replace the unbounded cones C_i with *trimmed* half-cones $D_i \subset \mathbb{R}^4$:

$$C_i := C(\mathbf{x}_i, r_i) = \{(\mathbf{x}, r) \in \mathbb{R}^4 \mid \|\mathbf{x} - \mathbf{x}_i\|_2 \leq |r - r_i|\},$$

$$D_i := D(\mathbf{x}_i, r_i) = C(\mathbf{x}_i, r_i) \cap (\mathbb{R}^3 \times [r_i - L, r_i]), \quad (11)$$

where $L > \max_{1 \leq i \leq n} r_i$ to guarantee that all displaced half-cones will intersect the $r = 0$ hyperplane. If we define $K_n := \bigcup_{1 \leq i \leq n} D_i$, the 3D solid $S_n(A)$ becomes a slice of the 4D solid $K_n(A)$ at $r = 0$, where $K_n : \mathbb{R}^{4n} \rightarrow \mathcal{P}(\mathbb{R}^4)$ is the discretization mapping illustrated in Fig. 3(d), (e). Unlike the scaled primitives B_i that have different sizes, their cones C_i and D_i are all translated instances of the same base shapes C_0 and D_0 , respectively, whose apexes are at the origin. Their union can thus be viewed as a Minkowski sum $K_n(A) = A \oplus D_0$. The 3D solid is then obtained as a 3D slice $S_n(A) = K_n(A)|_{r=0} := \gamma_0^{-1}(K_n(A) \cap \gamma_0(\mathbb{R}^3))$,⁷ illustrated in Fig. 3(e), (f).

To obtain an analytic form for the 4D geometry similar to (8), let $D_0 = U_0^*(f_{D_0})$, hence a defining function for each compact cone can be obtained as $f_{D_i} = f_{D_0}(\mathbf{a} - \mathbf{a}_i)$, where $\mathbf{a} = (\mathbf{x}, r) \in \mathbb{R}^4$. If we form an impulsive density function similar to (7)

$$\rho_A(\mathbf{a}) := \sum_{1 \leq i \leq n} c_i \delta_{\mathbf{a}_i}(\mathbf{a}) = \sum_{1 \leq i \leq n} c_i \delta^4(\mathbf{a} - \mathbf{a}_i), \quad (12)$$

where $\delta^4(\mathbf{a}) = \delta^3(\mathbf{x})\delta(r)$, the convolution in (8) generalizes to the 4-space as

$$f_{K_n(A)} = \rho_A * f_{D_0}, \quad \text{i.e., } A \oplus D_0 = U_0^*(\rho_A * f_{D_0}), \quad (13)$$

whose domain restriction to the $r = 0$ hyperplane gives a bump function for the 3D solid as $f_{S_n(A)} = f_{K_n(A)}|_{r=0}$,⁸ therefore $S_n(A) = U_0^*(f_{K_n(A)}|_{r=0})$. As before, we can let $f_{C_0}(\mathbf{x}, r) := \psi_\alpha(\|\mathbf{x}\|_p/r)$ whose unbounded support is smoothly trimmed along the 4th dimension to $r \in (-L, 0)$ as $f_{D_0}(\mathbf{x}, r) := \psi_\alpha(\|\mathbf{x}\|_p/r)\psi_\alpha(1 - 2r/L)$.

2.3. Sampling strategies

Clearly, a solid $S \in \mathcal{S}$ of arbitrary shape cannot in general be exactly constructed as a finite union of balls $S_n(A) = A \otimes \gamma_0(B_0)$, i.e., a 3D slice of $K_n(A) = A \oplus D_0$. However, a similar construction is

⁷ For a set $K \subset \mathbb{R}^4$, we use the simplified notation $K|_{r=r_0}$ for its $r = r_0$ slice projected to \mathbb{R}^3 , i.e., $K|_{r=r_0} := \gamma_0^{-1}(K \cap \gamma_0(\mathbb{R}^3))$, where $\gamma_r : \mathbb{R}^3 \hookrightarrow \mathbb{R}^4$ is defined as $\gamma_r(\mathbf{x}) = (\mathbf{x}, r)$ hence $\gamma_r^{-1}(\mathbf{x}, r) = \mathbf{x}$.

⁸ For a function $f_K : \mathbb{R}^4 \rightarrow \mathbb{R}$, we use the notation $f_K|_{r=r_0} : \mathbb{R}^3 \rightarrow \mathbb{R}$ to denote the restriction (and trivial projection) of its domain to the $r = r_0$ hyperplane, i.e., $f_K|_{r=r_0}(\mathbf{x}) = f_K(\mathbf{x}, r_0)$.

possible by replacing the finite set of ball centers P and its radius-lift A , with the MA (denoted by $\mathcal{M}[S]$) and the MAT (denoted by $\mathcal{T}[S]$), respectively. In fact $S = \mathcal{T}[S] \otimes^* \gamma_0(B_0)$, which is a 3D slice of $K = \mathcal{T}[S] \oplus^* D_0$.⁹ This construction can be thought of as sweeping a resizeable ball along the MA (with prespecified scaling for the balls along the MA trajectory), or equivalently, sweeping a rigid cone along the MAT in 4D followed by a 3D slicing. Unfortunately, the convolution formulation is not as simple in this case, because MA and MAT are not homogeneous, but are in general made of 2-, 1- or 0-dimensional subanalytic components for 3D solids [28]. We conjecture that it is possible to generalize the density function ρ_A defined in (12) to $\rho_{\mathcal{T}[S]}$, by using Dirac delta functions of various orders over different regions of $\mathcal{T}[S]$ depending on their dimensionalities, whose formal treatment we postpone to a follow-up study on skeletal density functions (SDF).

Here we take a simpler approach, by assuming sequences of finite samples $A \subset \mathbb{R}^4$ of different sizes $|A| = 1, 2, \dots$ that progressively approximate the shape. The set $S_n(A)$ is called an ϵ -approximation of S if $d_H(S_n(A), S) \leq \epsilon$, where d_H denotes the Hausdorff L^2 -metric. It is important to emphasize that our formulation in 2.2 does not impose any theoretical restriction on the sampling algorithm, as long as it guarantees that as $n \rightarrow \infty$, $S_n(A)$ converges to S (i.e., $\lim_{|A| \rightarrow \infty} d_H(S_n(A), S) = 0$), and the convolution in (13) holds in the limit for the *countably* infinite set of knots A . A variety of methods that have been in use in the CD literature [13–16] (Fig. 1(c)–(e)) can be used, two of which are briefly reviewed here due to their theoretical significance and computational relationship with our algorithm in Appendix A.

Hubbard [14] proposed an algorithm that populates the maximal balls over the MA (Fig. 1(d)), obtained from pruning the Voronoi diagram of a dense sampling over the boundary, and follows a principle of conservative coverage to create a bounding sphere-tree. In terms of our formulation, this is equivalent to selecting $A \subset \mathcal{T}(S)$ and has been shown in [14] to converge to the shape faster than octree-based sampling (Fig. 1(c)). However, MA and MAT are unstable with respect to C^0 - and C^1 - perturbations of the boundary [28], making their computations extremely difficult in the presence of noise/errors. Weller and Zachmann [16] proposed the inner-sphere tree (IST) method that precomputes the distance function over a uniform grid and uses a greedy algorithm to pack the interior of the solid, giving priority to the largest ball that fits at each step. This approach has been proven effective for real-time applications [16], but leaves out void spaces in the interior of the set that are undesirable for analytic modeling, and is nonoptimal for thin objects.

We use a similar greedy algorithm that is guided by the SDF field [34], which creates spherical samples that are similar to the outcomes of MA-based algorithms [14,15], without the need to compute and prune the MA, and is capable of producing better approximations than distance-based sphere packing [16] with fewer number of balls. Our algorithm guarantees bounds on the Hausdorff distance from the original shape that are proportional to the SDF grid resolution. This property is used in Section 5 as a basis for time complexity comparisons between operations on uniform samples versus spherical samples generated with the same input grid resolution. To prevent distraction from this article’s main focus on Minkowski discretizations and their Fourier reconciliations, the details of our spherical decomposition algorithm along with its topological properties and approximation error bounds are postponed to Appendix A.

⁹ The MA/MAT of an r -set are not necessarily closed, and neither are their Minkowski sums with a closed ball/cone, which is why the regularized Minkowski sum/product (denoted by \oplus^* / \otimes^*) need to be used.

2.4. Why spherical primitives?

The motion of rigid bodies can be abstracted as trajectories of points in the so-called ‘configuration space’ (commonly abbreviated as the \mathcal{C} -space), first introduced to the field of robotics by Lozano-Perez [3]. Every rigid motion in 3D can be parameterized by a tuple

$$M = (R, \mathbf{t}) \in SE(3), \quad SE(3) \cong SO(3) \times \mathbb{R}^3, \quad (14)$$

where $R \in SO(3)$ is a rotation described by a 3×3 special orthogonal matrix (i.e., $R^T = R^{-1}$ and $\det(R) = +1$) and $\mathbf{t} \in \mathbb{R}^3$ is a translation described by a 3-vector. See Appendix B for the definition, properties, and terminology of the motion group and its action on r -sets.

The advantage of using primitives with spherical symmetry becomes evident in the light of the isometric property of $SE(3)$. A 3D ball $B_0 := B(\mathbf{0}, r)$ is invariant under 3D rotations, i.e., $RB_0 = B_0$ hence $(R, \mathbf{t})B_0 = B(\mathbf{t}, r)$. The same invariance property can be asserted for 4D cones C_0 and D_0 whose axes stay parallel to the r -axis after 3D translations and rotations. Accordingly, the transformation of $S_n(A)$ and $K_n(A)$ amounts only to a relocation of the center and apex positions. For the equiradius case, the Minkowski sum in (8) for the transformed solid is given by

$$(R, \mathbf{t})S_n(P) = S_n((R, \mathbf{t})P) = ((R, \mathbf{t})P) \oplus B_0, \quad (15)$$

whose analytic expression is given by the bump function

$$f_{(R, \mathbf{t})S_n(P)} = \rho_{(R, \mathbf{t})P} * f_{B_0} = [\rho_P \circ (R, \mathbf{t})^{-1}] * f_{B_0}, \quad (16)$$

where the lumped density

$$\rho_{(R, \mathbf{t})P}(\mathbf{x}) = \sum_{1 \leq i \leq n} c_i \delta^3(\mathbf{x} - (R, \mathbf{t})\mathbf{x}_i), \quad (17)$$

is an implicit representation of the transformed set of 3D knots $(R, \mathbf{t})P = \{(R, \mathbf{t})\mathbf{x} \mid \mathbf{x} \in P\}$. In a similar fashion, for nonequiradius case the Minkowski sum in (13) for the transformed lifted geometry is given by

$$(R, \mathbf{t})K_n(A) = K_n((R, \mathbf{t})A) = ((R, \mathbf{t})A) \oplus D_0, \quad (18)$$

whose analytic expression is given by the bump function

$$f_{(R, \mathbf{t})K_n(A)} = \rho_{(R, \mathbf{t})A} * f_{D_0} = [\rho_A \circ (R, \mathbf{t})^{-1}] * f_{D_0}, \quad (19)$$

where the lumped density

$$\rho_{(R, \mathbf{t})A}(\mathbf{a}) = \sum_{1 \leq i \leq n} c_i \delta^4(\mathbf{a} - (R, \mathbf{t})\mathbf{a}_i), \quad (20)$$

is an implicit representation of the transformed set of 4D knots $(R, \mathbf{t})A = \{(R, \mathbf{t})\mathbf{a} \mid \mathbf{a} \in A\}$, using the trivial extension $(R, \mathbf{t})\mathbf{a} = ((R, \mathbf{t})\mathbf{x}, r)$ for $\mathbf{a} = (\mathbf{x}, r) \in \mathbb{R}^4$.

The strength of the discretization schemes that transform according to (15) through (20) lies in the rotation invariance of the primitive sets B_0 or D_0 and their radial kernels f_{B_0} or f_{D_0} , which appear in the same form in all equations before and after motion. We show in Section 3 that the same form is conserved across Minkowski sums and related cross-correlations between pairs of discretized objects. The practical implication is that the primitives do not need to take an explicit role in the computations, and numerical algorithms deal only with the discrete sets P or A or their density functions ρ_P or ρ_A .

2.5. Fourier expansions

A major motivation behind using analytic methods is the efficient tractability of convolutions and differentiations via

Fourier transforms. Using the orthonormal basis of the form $e^{2\pi i(\boldsymbol{\omega} \cdot \mathbf{x})}$ (where $i^2 = -1$), every r -set's bump function $f_S \in L^2(\mathbb{R}^3)$ can be expanded into its components given by $\hat{f}_S \in L^2(\mathbb{R}^3)$, using the continuous Fourier transform (CFT)¹⁰:

$$\hat{f}_S = \langle f_S, e^{+2\pi i(\boldsymbol{\omega} \cdot \mathbf{x})} \rangle = f_S = \langle \hat{f}_S, e^{-2\pi i(\boldsymbol{\omega} \cdot \mathbf{x})} \rangle. \quad (21)$$

For most applications such as the ones discussed in Section 4, the current analytic methods rely on discretizing the defining functions over a dense sample of points in the interior [17,18,22], which turns the CFT into a discrete Fourier transform (DFT). If the sampling is over a uniform grid, the DFT can be implemented very efficiently using the well-known fast Fourier transform (FFT) algorithm, first discovered by Cooley and Tukey [9]. The rotations are handled by an interpolation over the frequency grid and translations are embedded into convolutions. However, when the objects are discretized with spherical primitives whose centers form a nonuniform set of points, the CFT interpolates a nonequispaced DFT (NDFT), to which the classical FFT algorithms do not apply. Potts et al. [10] developed a nonequispaced FFT (NFFT) algorithm for efficient implementation of NDFT sums in asymptotically similar running times with the classical FFT. See Appendix C for the definition, properties, and complexity of Fourier transforms.

If an r -set $S \in \mathcal{S}$ moves via $(R, \mathbf{t}) \in SE(3)$, its transformed bump function changes in the frequency domain as $\hat{f}_{(R, \mathbf{t})S} = \zeta_{\mathbf{t}} \hat{f}_{RS} = \zeta_{\mathbf{t}} (\hat{f}_S \circ R^T)$, where $\zeta_{\mathbf{t}}(\mathbf{x}) := \delta(\mathbf{x} - \mathbf{t})$ denotes a shifted Dirac delta function that transfers to $\hat{\zeta}_{\mathbf{t}}(\boldsymbol{\omega}) = e^{-2\pi i(\boldsymbol{\omega} \cdot \mathbf{t})}$. For an equiradius discretization $S_n(P) = P \oplus B_0$ the Fourier expansion of $f_{S_n(P)} = \rho_P * f_{B_0}$ is a simple product $\hat{f}_{S_n(P)} = \hat{\rho}_P \hat{f}_{B_0}$. Applying the CFT to (16) gives

$$\hat{f}_{(R, \mathbf{t})S_n(P)} = \zeta_{\mathbf{t}} \hat{\rho}_P \hat{f}_{B_0} = \zeta_{\mathbf{t}} (\hat{\rho}_P \circ R^T) \hat{f}_{B_0}, \quad (22)$$

where the density function given in (7) is transferred to

$$\hat{\rho}_P(\boldsymbol{\omega}) = \sum_{1 \leq i \leq n} c_i \hat{\zeta}_{\mathbf{x}_i}(\boldsymbol{\omega}) = \sum_{1 \leq i \leq n} c_i e^{-2\pi i(\boldsymbol{\omega} \cdot \mathbf{x}_i)}. \quad (23)$$

The evaluation of (23) from a nonuniform set of 3D knots to a uniform 3D frequency grid amounts to a one-sided 3D NDFT computation.

For nonequiradius discretization $S_n(A) = K_n(A)|_{r=0}$ where $K_n(A) = A \oplus D_0$, we have $f_{K_n(A)} = \rho_A * f_{D_0}$ hence $\hat{f}_{K_n(A)} = \hat{\rho}_A \hat{f}_{D_0}$. Applying the CFT to (19) gives

$$\hat{f}_{(R, \mathbf{t})K_n(A)} = \zeta_{\mathbf{t}} \hat{\rho}_A \hat{f}_{D_0} = \zeta_{\mathbf{t}} (\hat{\rho}_A \circ R^T) \hat{f}_{D_0}, \quad (24)$$

where $\mathbf{a} = (\mathbf{t}, r) \in \mathbb{R}^4$ represents a lifted translation, and the density function given in (12) is transferred to

$$\hat{\rho}_A(\mathbf{v}) = \sum_{1 \leq i \leq n} c_i \hat{\zeta}_{\mathbf{a}_i}(\mathbf{v}) = \sum_{1 \leq i \leq n} c_i e^{-2\pi i(\mathbf{v} \cdot \mathbf{a}_i)}, \quad (25)$$

in which $\hat{\zeta}_{\mathbf{a}}(\mathbf{v}) = \zeta_{\mathbf{t}}(\boldsymbol{\omega}) e^{-2\pi i(\boldsymbol{\omega} \cdot \mathbf{r})}$ for the lifted physical domain $\mathbf{a}_i = (\mathbf{x}_i, r_i) \in A$ is obtained in a similar fashion to (23) using NDFTs, except that the frequency domain is also lifted to 4D as $\mathbf{v} = (\boldsymbol{\omega}, \eta) \in \mathbb{R}^4$.

3. Correlation functions

Having defined a spherical sampling in terms of a Minkowski sum of discrete knots and balls/cones alongside their analytic formulation in Section 2, we now investigate how they embed into cross-correlations between pairs of objects. In 3.1, we show that spherical discretization structure is preserved and carried into configuration pointsets, whose Fourier formulation is given in 3.2.

¹⁰ Bump functions are square-integrable, i.e., $C_0^\infty(\mathbb{R}^3) \subset L^2(\mathbb{R}^3)$.

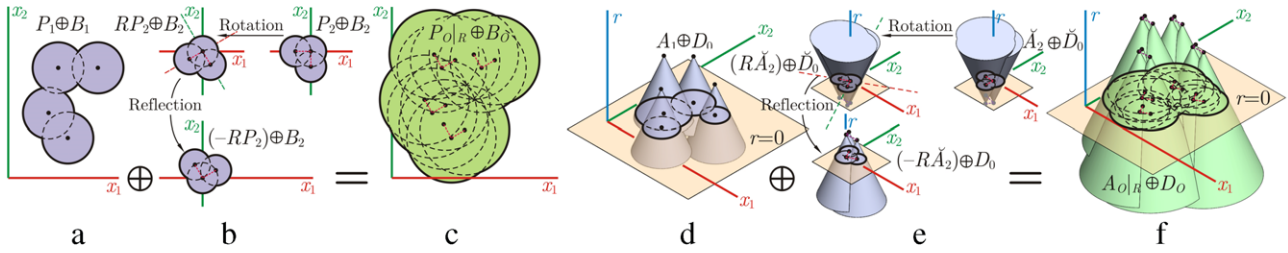


Fig. 4. The \mathcal{C} -obstacle is obtained as a Minkowski sum. The discretization scheme is closed under Minkowski sums for both equiradius and nonequiradius samples. For the latter, the summands must have the same cone orientation along the r -axis, requiring a pre-reflection.

3.1. Discrete correlations

Given two r -sets $S_1, S_2 \in \mathcal{S}$, we define their ‘correlation function’ as $g_{S_1, S_2} : SE(3) \times SE(3) \rightarrow \mathbb{R}$ where

$$g_{S_1, S_2}(M_1, M_2) = \langle f_{M_1 S_1}, f_{M_2 S_2} \rangle, \quad (26)$$

accumulates the pointwise multiplication of the overlapped shape descriptor functions of S_1 and S_2 moved using $M_1 = (R_1, \mathbf{t}_1)$ and $M_2 = (R_2, \mathbf{t}_2)$, respectively. The function in (26) can formulate, for example, a holonomic collision constraint (Section 4.1), a shape complementarity metric (Section 4.2), or a morphological operator (Section 4.3). For instance, comparing (26) with (4) shows that if $S_1 = U_0^*(f_{S_1})$ and $S_2 = U_0^*(f_{S_2})$, then $g_{S_1, S_2}(M_1, M_2)$ defines a collision predicate for the moved solids, i.e., $M_1 S_1 \cap^* M_2 S_2 \neq \emptyset$ iff $g_{S_1, S_2}(M_1, M_2) > 0$. It is easy to show that the correlation function only depends on the *relative* configuration $M := M_1^{-1} M_2$, i.e., the motion of S_2 observed from a coordinate frame attached to S_1 . Letting $M = (R, \mathbf{t}) = (R_1, \mathbf{t}_1)^{-1} (R_2, \mathbf{t}_2) = (R_1^T R_2, R_1^T (\mathbf{t}_2 - \mathbf{t}_1))$, the alternative formulation $g_{S_1, S_2} : SE(3) \rightarrow \mathbb{R}$ becomes

$$g_{S_1, S_2}(R, \mathbf{t}) = \langle f_{S_1}, (f_{RS_2} \circ T^{-1}) \rangle, \quad (27)$$

noting that $f_{(R, \mathbf{t})S} = f_{RS} \circ T^{-1} = (f_S \circ R^T) \circ T^{-1}$ where $T : \mathbb{R}^3 \rightarrow \mathbb{R}^3$, $T\mathbf{x} = \mathbf{x} + \mathbf{t}$. The inner product in (27) can be viewed as a 6D non-commutative convolution over $SE(3)$ [2]. For numerical tractability, we decompose the motion into rotational and translational parts, and view the latter as a 3D commutative convolution:

$$g_{S_1, S_2}(R, \mathbf{t}) = (f_{RS_2} \star f_{S_1})(\mathbf{t}) = (f_{S_1} \star f_{-RS_2})(\mathbf{t}), \quad (28)$$

where $-S = \{-\mathbf{x} \mid \mathbf{x} \in S\}$ denotes a reflection with respect to the origin. This defines a *relative* collision predicate, i.e., $S_1 \cap^* (R, \mathbf{t})S_2 \neq \emptyset$ iff $g_{S_1, S_2}(R, \mathbf{t}) > 0$, and brings us to the important concept of a ‘configuration obstacle’ (or the \mathcal{C} -obstacle for short) in robotics [3], defined as

$$O_{S_1, S_2} = \{(R, \mathbf{t}) \in SE(3) \mid g_{S_1, S_2}(R, \mathbf{t}) > 0\}, \quad (29)$$

whose complement cO_{S_1, S_2} is the ‘free space’. The obstacle space does not contain the contact space by definition, i.e., is an open set, whose closure $O_{S_1, S_2}^* := \kappa O_{S_1, S_2}$ is a 6D r -set [22]. For a fixed rotation $R \in SO(3)$ the translational obstacle defined by

$$O_{S_1, S_2}|_R := \{\mathbf{t} \in \mathbb{R}^3 \mid g_{S_1, S_2}(R, \mathbf{t}) > 0\} \quad (30)$$

is a 3D slice through the 6D obstacle, whose closure is obtained by offsetting S_1 with $-RS_2$ given by a Minkowski sum

$$O_{S_1, S_2}^*|_R = S_1 \oplus (-RS_2) = U_0^*(f_{S_1} \star f_{-RS_2}). \quad (31)$$

As a result of the definition in (29), $g_{S_1, S_2}(R, \mathbf{t})$ serves as a defining function of the regularized \mathcal{C} -obstacle as a 0-sublevel set $O_{S_1, S_2}^* = U_0^*(g_{S_1, S_2})$, referred to as the ‘gap function’ [3]. Similarly, the restriction of the gap function to a fixed $R \in SO(3)$ is a defining function for the 3D slice $O_{S_1, S_2}^*|_R = U_0^*(g_{S_1, S_2}|_R)$, where $g_{S_1, S_2}|_R = f_{S_1} \star f_{-RS_2}$. Next we investigate the discretization of the \mathcal{C} -obstacle (and the

Minkowski sums in general) when spherical sampling is used for the constituent parts.

Equiradius correlations. First, let $S_1 = S_{n_1}(P_1)$ and $S_2 = S_{n_2}(P_2)$ be composed of instances of balls denoted by $B_1 := B(\mathbf{0}, r_1)$ and $B_2 := B(\mathbf{0}, r_2)$, respectively. Substituting from (15) in (31), and noting the invariance of the balls to reflection and rotation (i.e., $-RB_2 = B_2$) and the commutativity of Minkowski sums, we obtain

$$O_{S_1, S_2}^*|_R = (P_1 \oplus B_1) \oplus ((-RP_2) \oplus B_2) \quad (32)$$

$$= (P_1 \oplus (-RP_2)) \oplus (B_1 \oplus B_2) \quad (33)$$

$$= P_{0|R} \oplus B_0. \quad (34)$$

The first term $P_{0|R} := P_1 \oplus (-RP_2)$ is a finite set of $n_1 n_2$ points in the 3D translation space obtained from pairwise summations of ball centers in P_1 and $-RP_2$. It represents the discrete translational obstacle $O_{P_1, P_2}|_R$, a 3D slice through $P_0 := O_{P_1, P_2} \subset SE(3)$ defining a collection of $n_1 n_2$ curves. The second term $B_0 := B_1 \oplus B_2 = B(\mathbf{0}, r_0)$ is a ball of radius $r_0 = r_1 + r_2$ in the translational space, representing the ‘primitive obstacle’ $O_{B_1, B_2}|_R$, which is a 3D slice of $O_{B_1, B_2} \subset SE(3)$. Therefore, the total obstacle itself is a finite union of balls, i.e., discretized with the same scheme as the original objects: $O_{S_1, S_2}^*|_R = S_{n_1 n_2}(P_{0|R})$. An illustration is given in Fig. 4(a)–(c). The analytic formulation in (28) develops in parallel as

$$g_{S_1, S_2}|_R = (\rho_{P_1} \star f_{B_1}) \star (\rho_{-RP_2} \star f_{B_2}) \quad (35)$$

$$= (\rho_{P_1} \star \rho_{-RP_2}) \star (f_{B_1} \star f_{B_2}), \quad (36)$$

$$= \rho_{P_{0|R}} \star f_{B_0}, \quad (37)$$

where $\rho_{P_{0|R}} := (\rho_{P_1} \star \rho_{-RP_2})$ is the impulsive density function of the discrete pointset $P_{0|R}$ made of $n_1 n_2$ impulses corresponding to pairwise cross-correlation of Dirac delta terms from the constituents,¹¹ while $f_{B_0} := f_{B_1} \star f_{B_2}$ is a cross-correlation of two radial bumps of radii r_1 and r_2 , leading to another radial bump of radius $r_0 = r_1 + r_2$ that defines the obstacle ball B_0 . Note that if we choose $f_{B_1}(\mathbf{x}) := \psi_\alpha(\|\mathbf{x}\|_2/r_1)$ and $f_{B_2}(\mathbf{x}) := \psi_\alpha(\|\mathbf{x}\|_2/r_2)$ using the form in (10), their convolution does not take the same form, i.e., $f_{B_0}(\mathbf{x}) \neq \psi_\alpha(\|\mathbf{x}\|_2/r_0)$. However, the latter can be safely replaced for the last term in (37) without changing the obstacle $O_{S_1, S_2}^*|_R = U_0^*(g_{S_1, S_2}|_R)$, since the bump form choice is arbitrary as long as $B_0 = U_0^*(f_{B_0})$.

Nonequiradius correlations. The generalization to $S_1 = S_{n_1}(A_1)$ and $S_2 = S_{n_2}(A_2)$ made of primitive balls of different sizes is not straightforward. This is because the commutativity of the Minkowski sum $S_n(P) = P \oplus B_0$ that led from (32) to (33) does not hold for the product $S_n(A) = A \otimes \gamma_0(B_0)$. In terms of the lifted geometry, this manifests as the observation that the 3D Minkowski sum of the cross-sections is not equal to the cross-section of the 4D Minkowski sum; or in other words, a collision between $K_1 = K_{n_1}(A_1)$ and $K_2 = K_{n_2}(A_2)$ does not necessarily

¹¹ Note that if $\rho_1(\mathbf{x}) := \delta^3(\mathbf{x} - \mathbf{x}_1)$, $\rho_2(\mathbf{x}) := \delta^3(\mathbf{x} - \mathbf{x}_2)$, and $\tilde{\rho}(\mathbf{x}) := \rho(-\mathbf{x})$, then $(\rho_2 \star \rho_1)(\mathbf{t}) = (\rho_1 \star \tilde{\rho}_2)(\mathbf{t}) = \delta^3(\mathbf{t} - (\mathbf{x}_1 + \mathbf{x}_2))$.

imply a collision between the cross-sections $S_1 = K_{n_1}(A_1)|_{r=0}$ and $S_2 = K_{n_2}(A_2)|_{r=0}$. At the primitive level, this is because the 4D half-cones, despite being invariant under 3D rotations and reflections, are *not* invariant under 4D reflections, hence $D_0 \neq -D_0$ and the sum $D_0 \oplus (-D_0)$ (that appears in $K_1 \oplus (-RK_2)$) does *not* give a half-cone in the \mathcal{E} -space. Fortunately, this can be solved by a pre-reflection with respect to the $r = 0$ hyperplane of one of the two lifted shapes. If we let

$$\check{K} = \{(\mathbf{x}, -r) \mid (\mathbf{x}, r) \in K\}, \quad (K, \check{K} \subset \mathbb{R}^4) \quad (38)$$

denote the r -mirror image of the 4D set K , the 3D set can be retrieved from both of them as $S = K|_{r=0} = \check{K}|_{r=0}$. Then the nonequiradius discretization scheme in (18) gives $\check{K}_n(\check{A}) = \check{A} \oplus \check{D}_0$ and $-\check{K}_n(\check{A}) = (-\check{A}) \oplus D_0$, noting that $D_0 = -\check{D}_0$ and the sum $D_0 \oplus (-\check{D}_0)$ (that appears in $K_1 \oplus (-RK_2)$) is a half-cone of double the size in the \mathcal{E} -space. Furthermore, it is easy to prove that in this case, for two collections of half-cones of *opposite* directions that intersect the $r = 0$ hyperplane, a collision between the 4D solids does in fact imply a collision between the 3D slices:

Lemma 2. $S_1 \cap^*(R, \mathbf{t})S_2 \neq \emptyset \Rightarrow K_1 \cap^*(R, \mathbf{t})\check{K}_2 \neq \emptyset$.

Proof. The proof is straightforward, by noting that for every pair of 4D cones $D_1 := D(\mathbf{x}_1, r_1)$ and $D_2 := D(\mathbf{x}_2, r_2)$ corresponding to $(\mathbf{x}_1, r_1) \in A_1$ and $(\mathbf{x}_2, r_2) \in A_2$, respectively, they intersect after flipping one of them upside down (i.e., $D_1 \cap^* \check{D}_2 \neq \emptyset$) if and only if their 3D slices intersect (i.e., $D_1|_{r=0} \cap^* \check{D}_2|_{r=0} \neq \emptyset$). The assertion is very easy to picture for 3D cones whose slices are 2D disks. \square

As a direct corollary, we can define a 4D translational \mathcal{E} -obstacle that is discretized with the same scheme as

$$O_{K_1, \check{K}_2}^*|_R = (A_1 \oplus D_0) \oplus ((-R\check{A}_2) \oplus D_0) \quad (39)$$

$$= (A_1 \oplus (-R\check{A}_2)) \oplus (D_0 \oplus D_0) \quad (40)$$

$$= A_0|_R \oplus D_0, \quad (41)$$

and the 3D obstacle is a slice $O_{S_1, S_2}^*|_R = [O_{K_1, \check{K}_2}^*|_R]_{r=0}$. The first term $A_0|_R := A_1 \oplus (-R\check{A}_2)$ is a finite set of $n_1 n_2$ points in the 4D translation space obtained from pairwise summations of cone apexes in A_1 and $-R\check{A}_2$, which is the same as $O_{A_1, \check{A}_2}^*|_R$. In this case, the primitive obstacle $D_0 := D_0 \oplus D_0$ is a larger half-cone with a height of $2L$, which is equal to $O_{D_0, D_0}^*|_R$. Therefore, the \mathcal{E} -obstacle is discretized with the same scheme as $O_{S_1, S_2}^*|_R = S_{n_1 n_2}(A_0|_R) = [K_{n_1 n_2}(A_0|_R)]_{r=0}$. An illustration is given in Fig. 4(d)–(f). The analytic formulation in (28) develops in parallel as

$$g_{K_1, \check{K}_2}|_R = (\rho_{A_1} * f_{D_0}) * (\rho_{-R\check{A}_2} * f_{D_0}) \quad (42)$$

$$= (\rho_{A_1} * \rho_{-R\check{A}_2}) * (f_{D_0} * f_{D_0}) \quad (43)$$

$$= \rho_{A_0|_R} * f_{D_0}, \quad (44)$$

whose restriction to $r = 0$ gives $g_{S_1, S_2}|_R = [g_{K_1, \check{K}_2}|_R]_{r=0}$. Similar to the equiradius case, $\rho_{A_0|_R} := (\rho_{A_1} * \rho_{-R\check{A}_2})$ is the impulsive density function of the discrete pointset $A_0|_R$ made of $n_1 n_2$ impulses corresponding to cross-correlation of pairs of shifted Dirac delta functions, while $f_{D_0} := (f_{D_0} * f_{D_0})$ is an auto-correlation, which can be arbitrarily modified from the original convolved form to $f_{D_0}(\mathbf{t}, r) := \psi_\alpha(\|\mathbf{t}\|_2/r)\psi_\alpha(1 - r/L)$ without changing the obstacle $O_{S_1, S_2}^*|_R = U_0^*(g_{K_1, \check{K}_2}|_{r=0})$.

Although one only needs the $r = 0$ slice to retrieve the \mathcal{E} -obstacle, the other slices carry useful information. In fact, any $r \neq 0$ slice corresponding to $r \in (-L, +L)$ gives the obstacle for a pair of *offset* 3D solids defined as r_1 -slice of K_1 (i.e., shrinking S_1 's primitives by r_1) and r_2 -slice of \check{K}_2 (i.e., expanding S_2 's primitives by r_2) giving a total offset of $-r = -(r_1 + r_2)$. These 'offset obstacles' can be used, for example, to incorporate tolerances for

machine tooling, guarantee safety margins for path planning, or construct skin layers for protein shape complementarity modeling (Section 4).

3.2. Fourier correlations

To take advantage of the convolution theorem for a significantly faster computation of correlation functions, we present an analysis of the gap function in the Fourier domain. For two r -sets $S_1, S_2 \in \mathcal{X}$, applying CFT to (28) yields the Fourier correlation function as

$$\hat{g}_{S_1, S_2}|_R = \hat{f}_{S_1} \tilde{f}_{RS_2} = \hat{f}_{S_1} (\tilde{f}_{S_2} \circ R^T), \quad (45)$$

noting that $\hat{f}_{-RS} = \tilde{f}_{RS}$ for real defining functions, i.e., CFT converts reflection (in both physical and frequency domains) to conjugation in the frequency domain, a property known as Hermitian symmetry (see Appendix C). $\hat{g}_{S_1, S_2}|_R = \mathcal{F}\{g_{S_1, S_2}|_R\}$ is the CFT of $g_{S_1, S_2}(R, \mathbf{t})$ only with respect to translation at a fixed $R \in SO(3)$.

For the equiradius discretizations $S_{n_1}(P_1) = P_1 \oplus B_1$ and $S_{n_2}(P_2) = P_2 \oplus B_2$ with Fourier representations $\hat{f}_{S_{n_1}(P_1)} = \hat{\rho}_{P_1} \hat{f}_{B_1}$ and $\hat{f}_{S_{n_2}(P_2)} = \hat{\rho}_{P_2} \hat{f}_{B_2}$, respectively, their Fourier correlation is obtained by substituting (22) in (45), or directly applying the CFT to (37) as

$$\hat{g}_{S_1, S_2}|_R = (\hat{\rho}_{P_1} \hat{f}_{B_1}) (\tilde{\rho}_{RP_2} \tilde{f}_{B_2}) \quad (46)$$

$$= (\hat{\rho}_{P_1} \tilde{\rho}_{RP_2}) (\hat{f}_{B_1} \hat{f}_{B_2}) = \hat{\rho}_{P_0|_R} \hat{f}_{B_0}, \quad (47)$$

noting that $f_{B_{1,2}} = f_{-B_{1,2}}$ hence $\hat{f}_{B_{1,2}} = \tilde{f}_{B_{1,2}}$ (i.e., are both real-valued), and so is $\hat{f}_{B_0} = \hat{f}_{B_1} \hat{f}_{B_2}$. As expected, $\hat{\rho}_{P_0|_R} = \hat{\rho}_{P_1} \tilde{\rho}_{RP_2}$ is computed from a pointwise multiplication of the NDFTs over the knots P_1 and P_2 .

Analogously, for the nonequiradius discretizations $S_{n_1}(A_1) = K_{n_1}(A_1)|_{r=0}$ and $S_{n_2}(A_2) = \check{K}_{n_2}(\check{A}_2)|_{r=0}$, where $K_{n_1}(A_1) = A_1 \oplus D_0$ and $\check{K}_{n_2}(\check{A}_2) = \check{A}_2 \oplus \check{D}_0$ with $\hat{f}_{K_{n_1}(A_1)} = \hat{\rho}_{A_1} \hat{f}_{D_0}$ and $\hat{f}_{\check{K}_{n_2}(\check{A}_2)} = \hat{\rho}_{\check{A}_2} \hat{f}_{\check{D}_0}$, respectively, their Fourier correlation is obtained by substituting (24) in (45), or directly applying CFT to (44) as

$$\hat{g}_{K_1, \check{K}_2}|_R = (\hat{\rho}_{A_1} \hat{f}_{D_0}) (\tilde{\rho}_{R\check{A}_2} \tilde{f}_{\check{D}_0}) \quad (48)$$

$$= (\hat{\rho}_{A_1} \tilde{\rho}_{R\check{A}_2}) (\hat{f}_{D_0} \hat{f}_{\check{D}_0}) = \hat{\rho}_{A_0|_R} \hat{f}_{D_0}, \quad (49)$$

where $\hat{f}_{D_0} = \hat{f}_{D_0}^2 = \hat{f}_{\check{D}_0}^2$, noting that $\hat{f}_{D_0} = \tilde{f}_{\check{D}_0}$ as a result of the reflective duality $f_{D_0} = f_{-\check{D}_0}$. In a similar fashion, $\hat{\rho}_{A_0|_R} = \hat{\rho}_{A_1} \tilde{\rho}_{R\check{A}_2}$ is computed from a pointwise multiplication of the NDFTs over the knots A_1 and \check{A}_2 .

A critical observation is that the computational implementation relies only on the discrete knots P and A (or \check{A}), expressed in the Fourier domain by the NDFTs $\hat{\rho}_P$ in (23) and $\hat{\rho}_A$ (or $\hat{\rho}_{\check{A}}$) in (25), respectively. The continuous geometry is completely embodied by the primitives implicit in $\hat{f}_{B_0} = \hat{f}_{B_1} \hat{f}_{B_2}$ or $\hat{f}_{D_0} = \hat{f}_{D_0}^2 = \hat{f}_{\check{D}_0}^2$. However, despite appearing in equations, they do *not* explicitly participate into the numerical algorithms, and that reflects the true power of this particular discretization scheme.

4. Applications

Next, we show how the tools developed in Sections 2 and 3 can be applied to collision detection in 4.1, shape complementarity in 4.2, and configuration products in 4.3, and identify future research opportunities in each area.

4.1. Collision detection

Analytic collision detection (CD) can be traced to the work by Comba [29] on convex sets. Kavraki [17] discovered the interpretation of the translational \mathcal{C} -obstacle as a convolution of the objects – the robot and its workspace in the context of path planning [3] – along with the application of the FFT. Both objects $S_1, S_2 \in \mathcal{A}$ are represented by binary indicators $\mathbf{1}_{S_1}, \mathbf{1}_{S_2} : \mathbb{R}^3 \rightarrow \{0, 1\}$, discretized as bitmaps, and the integer map of the translational \mathcal{C} -obstacle obtained as $g_{S_1, S_2}(\mathbf{t}) = (\mathbf{1}_{S_1} * \mathbf{1}_{-S_2})(\mathbf{t})$ simply counts the number of grid cells that overlap at a relative translation $\mathbf{t} \in \mathbb{R}^3$. The algorithm performs two forward FFTs to obtain $\hat{\mathbf{1}}_{S_1}$ and $\hat{\mathbf{1}}_{S_2}$, a pairwise multiplication to obtain $\hat{g}_{S_1, S_2} = \hat{\mathbf{1}}_{S_1} \hat{\mathbf{1}}_{S_2}$, and an inverse FFT to retrieve the obstacle map in $O(m \log m)$ time, where m is the grid size. Although the algorithm is asymptotically optimal to obtain a complete description of the obstacle for all possible translations in a given discretized domain, it is rarely useful for time-critical CD (e.g., in real-time simulations and physically-based modeling [16]) where a *single* configuration is queried.

Lysenko [22] recently generalized the approach by using bump functions to facilitate differentiation, and proposed techniques to enable time-critical CD for a single-configuration query via truncated Fourier expansions, along with an analytic groundwork for early-hit/miss tests. Noting that the inner product structure is preserved by the CFT according to Parseval's theorem (See Appendix C), the collision predicate in (28) for a single relative configuration $(R, \mathbf{t}) \in SE(3)$ can be obtained as

$$g_{S_1, S_2}|_R(\mathbf{t}) = \left\langle f_{S_1}, (f_{RS_2} \circ T^{-1}) \right\rangle = \left\langle \hat{f}_{S_1}, \hat{c}_t \hat{f}_{RS_2} \right\rangle, \quad (50)$$

noting that $f_{(R, \mathbf{t})S} = f_{RS} \circ T^{-1} = (f_S \circ R^T) \circ T^{-1}$ which transforms to $\hat{f}_{(R, \mathbf{t})S} = \hat{c}_t \hat{f}_{RS} = \hat{c}_t (\hat{f}_S \circ R^T)$. As mentioned earlier, $T^{-1}\mathbf{x} := \mathbf{x} - \mathbf{t}$ is the shift function whose Fourier operator $\hat{c}_t(\boldsymbol{\omega}) = e^{-2\pi i(\boldsymbol{\omega} \cdot \mathbf{t})}$ is the CFT of the shifted Dirac Delta $c_t(\mathbf{x}) = (\delta^3 \circ T^{-1})(\mathbf{x}) = \delta^3(\mathbf{x} - \mathbf{t})$.

If a grid-based discretization is used, the rotation can be incorporated by a trilinear interpolation in either domain. Although a brute-force computation of the physical domain inner product over a grid of size m takes $O(m)$ – without a simple way of reducing the complexity once the grid resolution is fixed upfront – the frequency domain integral can be computed in $O(m')$ over a truncated grid of much smaller size $m' \ll m$ specified on-the-fly. This provides a mechanism for trading off accuracy with time, by a spiral traversal of the frequency grid starting from the dominant modes until the available time is over. On the other hand, the numerous combinatorial CD methods developed over the years (reviewed in [21]) exploit a variety of data structures to avoid brute-force testing in the physical domain, the likes of which are not available in the frequency domain. The sphere-tree methods [13–16] are among the most efficient, which enable another trade-off mechanism by descending down the tree until the time allocated to CD is consumed. Our framework enables exploiting the existing combinatorial techniques alongside the recent analytic methods in both domains. The details pertaining to the following are beyond the scope of this article and will be presented elsewhere: (1) *early-hit test* by limiting the integration of (50) to an intersection with a ball, which is a simple multiplication in the physical domain via (3); (2) *early-miss test* by offsetting (i.e., Minkowski sum) with a single ball, which is a simple multiplication in the frequency domain; and (3) *differentiation* of (50) for contact force/torque computation, using pairwise spherical primitive interactions.

4.2. Shape complementarity

Surface shape complementarity (SC) is a predominant determinant of successful binding of protein molecules, and is critical in

early-stage lead compound generation for rational drug design. The numerous FFT-based correlation techniques developed over the years (reviewed in [23]) use the same principles as analytic CD or path planning, except that they quantify SC by overlapping skin-layers. The so-called 'double-skin layer' approach [7] integrates the skin–skin intersections to obtain a SC 'score function' and subtracts core–core collisions as penalty, which add up to a convolution of 'affinity functions' of individual molecules (analogous to defining functions for CD, except with different complex-valued weights for skin/core atoms). Chen and Weng [18] described successful heuristics for weight assignment rasterized on a uniform grid along with the use of FFT. Bajaj et al. [7] proposed a faster grid-free method along with the use of NFFT, which has been highly influential in the development of our ideas.

For SC analysis of arbitrary shapes with important applications in assembly automation, packaging and nesting, and path planning in narrow environments, in addition to protein docking, we propose a reformulation of the double-skin layer approach by defining the complex affinity function $F_S : \mathbb{R}^3 \rightarrow \mathbb{C}$ for an arbitrary r -set $S \in \mathcal{A}$ as

$$F_S(\mathbf{x}) := i (f_{S \oplus B_0}(\mathbf{x}) - \lambda f_S(\mathbf{x})), \quad (51)$$

where $f_S, f_{S \oplus B_0} \in C_0^\infty(\mathbb{R}^3)$ are bump functions of the shape and its offset by the ball $B_0 = B(\mathbf{0}, r_0)$. The offset $r_0 > 0$ is decided depending on the feature size (e.g., set to the size of a water molecule (1.4 Å) for protein docking [7], or to MA-based local/weak feature size [28] for other applications), and $\lambda > 0$ defines the 'penalty factor'. For two shapes $S_1, S_2 \in \mathcal{A}$, the cross-correlation of their affinity functions gives the SC score as

$$G_{S_1, S_2}(R, \mathbf{t}) = (F_{RS_2} \star F_{S_1})(\mathbf{t}) = (F_{S_1} \star \bar{F}_{-RS_2})(\mathbf{t}). \quad (52)$$

Substituting for F_S and noting that $f_{S \oplus B_0} = f_S \star f_{B_0}$:

$$G_{S_1, S_2}|_R = \lambda^2 g_{S_1, S_2}|_R - 2\lambda g_{S_1, S_2}|_R \star f_{B_0} + g_{S_1, S_2}|_R \star f_{B_0}, \quad (53)$$

where the terms $g_{S_1, S_2}|_R = f_{RS_2} \star f_{S_1} = f_{S_1} \star f_{-RS_2}$ and $f_{B_0} = f_{B_0} \star f_{B_0}$ were studied earlier. If $\lambda > 1$ is large enough, the first term on the right-hand side of (53) takes interior–interior collisions into account with a penalty of $\propto -O(\lambda^2)$, the second term includes offset–interior overlaps with a reward of $\propto +O(\lambda)$, while the third term adds a smaller penalty of $\propto -O(1)$ for offset–offset overlaps. An important observation is that for spherical sampling, the offsets correspond to an increase of radii in the primitive balls, which in turn corresponds to a 3D slice of *elevated* cones in 4D (i.e., translation of the knots $A \in \mathbb{R}^4$ along the r -axis). The different terms in (53) thus become 3D slices of $g_{K_1, K_2}|_R$ corresponding to $r = 0, r_0$, and $2r_0$, all of which can be cumulatively obtained from a single 4D NDFT. The time-critical query in (50) for CD can also be extended to SC formulation, useful in real-time energy computations for interactive assembly or protein docking. More elaboration on these and other possibilities are left to a follow-up publication, including a study of *offset sensitivity* defined as differentiation of $g_{K_1, K_2}|_R$ with respect to r and its implications for SC analysis—think of the infinitesimal overlaps in (53) when $\lambda = 1$ and $r_0 \rightarrow 0^+$.

4.3. Morphological operations

Roerdink [1] generalized the concept of Minkowski sums/differences to Minkowski products/quotients over general groups, whose noncommutative convolutional formulation was presented by Lysenko et al. [2]. Nelaturi and Shapiro [4] applied the concept to $SE(3)$ (in this context referred to as 'configuration products/quotients') and showed its applicability to direct and inverse \mathcal{C} -space problems ranging from computing general sweeps to solving for maximal shapes and motions subject to containment

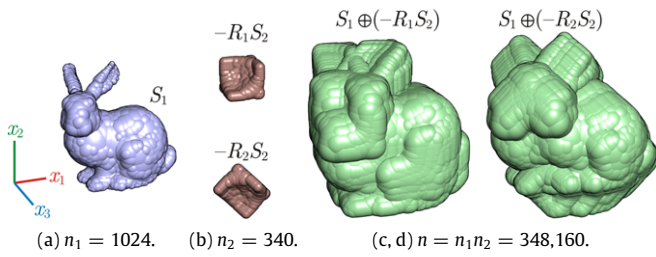


Fig. 5. The Minkowski sum of two spherical samples (a), (b) for two sampled rotations, which are sections through the 6D Minkowski product, obtained from pairwise Minkowski sum of primitives (c), (d).

constraints. The method embeds the solids in $SE(3)$, and uses a uniform sampling over translations and rotations followed by pairwise matrix multiplications across the two samples to compute the \mathcal{C} -products and quotients. Our discretization scheme can readily be applied to more efficiently sample the translation space for different rotational sections through the 6D domain, as will be demonstrated in Section 5.1.

An interesting extension of the method would be to formulate 6D spherical sampling of the subsets of the Riemannian manifold $SE(3)$ based on geodesic distances (see Appendix B for more details). One possible application is in machine tool path planning in the presence of tolerances [35], where the embedded workpiece complements the configuration product of the motion trajectory and tool profile. The tolerances can be introduced into either set by Minkowski operations between the ‘nominal’ geometry and primitive tolerance sets, e.g., Euclidean (for translational tolerances) and geodesic (for rotational tolerances) disks, cylinders, balls, or tori, all of which can be more efficiently discretized via spherical primitives than uniform samples. In a similar fashion to the constructions in Section 3.1, the tool’s swept volume is then given by a 3D projection of the 6D Minkowski product of the embedded tool profile and its motion, each of which are described by a Minkowski product of sample points on their nominal sets with the primitive tolerance sets. Rearranging the terms (similar to (32) through (34)) abstracts the tolerances away into a 6D configuration space tolerance set, and allows working with lower-dimensional nominal sets only. Unfortunately, the corresponding Fourier analysis in this case becomes quite tedious, whose potential benefits are unclear at this stage. See the example in [35] for an elaborate discussion.

A more important research question remains regarding the extension of spherical sampling to dual operations, i.e., Minkowski differences or quotients, which would open up the opportunity to extend the benefits of this approach to inverse problems in configuration modeling.

5. Numerical results

In this section we demonstrate how spherical sampling outperforms uniform sampling for Minkowski computations that are central to the range of applications discussed in Section 4, and validate the additional performance improvement by using FFT algorithms. We implemented the method as a C++ API that reads triangular meshes, generates spherical decompositions (using Algorithm 1 in Appendix A), converts the geometry to an analytic representation in the physical and/or frequency domains, and computes the correlations in either domain. We report on both CPU- and GPU-parallel computing, implemented using C++ Boost [36] and CUDA-C libraries, respectively. Our numerical experiments were conducted on a desktop computer with Intel Xeon E5-2687W CPU (32 cores, 3.10 GHz clock-rate, 64 GB host memory) and NVIDIA Tesla K20c GPU (2496 CUDA cores, 5 GB device memory).

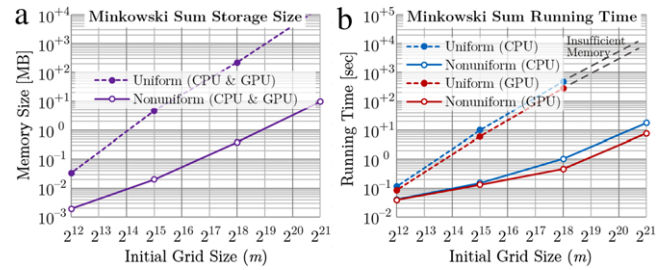


Fig. 6. Nonuniform spherical sampling significantly outperforms uniform sampling, by efficient use of memory and time resources.

5.1. Combinatorial advantage

Fig. 5 illustrates the \mathcal{C} -obstacle construction discussed in Section 3.1 for a pair of objects discretized via nonequiradius spherical sampling developed in Section 2.2. We repeat Algorithm 1 with different grid sizes m , and compare the arithmetic complexity of the result with that of the uniform sampling of the same grid dimensions used in [4]. As described in Appendix A, this guarantees that the Hausdorff metric-based approximation error of the spherical discretization is upperbounded by that of the uniform sample, which is $\epsilon = \sqrt{3}(L/m^{1/3})$. Therefore, the ‘initial grid size’ m will be used as a measure of resolution for both methods for the purpose of comparison.

As reported in Table 1, our method offers a clear advantage, decreasing the complexity by several orders of magnitude. Fig. 6(a), (b) plots the memory requirement and running times, respectively, for Minkowski sum computation (viewed as a translational cross-section of the configuration product, as described in Section 4.3) by pairwise summations in the nonuniform 4D sample space, compared to pairwise summations in the uniform 3D sample space used in [4]. The speed-ups of our method scale significantly with resolution, and reaches the range 400–600 \times (on both CPU and GPU)¹² for a grid size of $m := 2^{18} = 262144$, decreasing the CPU/GPU running times from 28/48 s to 104/47 ms. For larger initial grid sizes such as $m := 2^{21} = 2097152$, the memory cannot accommodate the Minkowski sum of uniform samples, while our method succeeds and carries out the sum in less than a second.

5.2. Analytic advantage

We next consider the computational performance of the analytic method, using both uniform and nonuniform sampling. Given the spherical decomposition of the two solids, their bump functions as a sum of radial kernels can be rasterized on uniform grids in the physical domain. The gap function representation of the Minkowski sum can then be computed by two forward FFTs, a pointwise multiplication over the frequency grids, and an inverse FFT to retrieve the result, whose running times are separately plotted in Fig. 7(g). The GPU implementation in this case offers significant speed-ups of 400–800 \times over its CPU counterpart.¹³ Comparing the results with Fig. 6(b) shows an improvement of 30–80 \times over the pairwise computations in the physical domain. For a grid size of $m := 2^{21} = 2097152$, accurate computation of the convolution takes less than 80 ms on the GPU.

¹² In each individual scenario, the GPU runs are only slightly faster than CPU runs (1–3 \times) due to extensive global memory references, but can be improved in future versions by memory optimization.

¹³ The FFT is implemented using FFTW [37] on the CPU and using cuFFT(W) on the GPU. With the exception of FFTW, all other CPU and GPU routines were written in parallel.

Table 1

Comparison of the sample size between grid-based uniform and grid-free spherical samplings. The ratio scales rapidly with size.

m	Uniform sampling			Spherical sampling			Ratio n'/n
	n'_1	n'_2	$n' = n'_1 n'_2$	n_1	n_2	$n = n_1 n_2$	
2^{12}	666	44	29,304	49	26	1,274	23.0
2^{15}	5,921	689	4.08×10^6	159	83	13,197	309.1
2^{18}	49,981	3,867	1.93×10^8	1024	340	3.48×10^5	764.5
2^{21}	409,058	36,874	1.5×10^{10}	3686	1081	3.98×10^6	3785.5

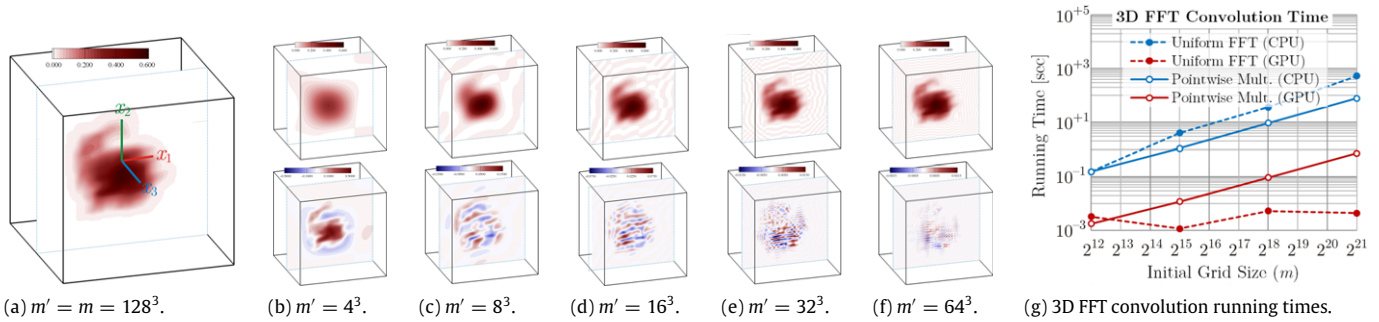


Fig. 7. A section through the gap function representation (a) of the Minkowski sum in Fig. 5; its approximations (b)–(f) with truncated Fourier expansions (top), and the residual error (bottom). Uniform grid-based FFT implementation of the convolution outperforms the pairwise primitive multiplication method by two orders of magnitude (g).

It appears that the pointwise multiplication step is the bottleneck in the FFT-based convolutions. However, by performing this step (and the following inverse FFT) over a small subset of size $m' \ll m$ of the frequency grid in the neighborhood of the dominant modes, one could decide on the amount of computation time to spend in a trade-off with accuracy depicted in Fig. 7(a)–(f). It is clear that small gap function errors do not necessarily imply small geometric discrepancies of the 0-sublevel set in terms of Hausdorff metric. However, it was shown by Lysenko [22] that it is also possible to impose upperbounds on the Hausdorff distance-based error as a function of the number of retained frequencies.

As depicted in Sections 2.5 and 3.2, the uniform 3D grid-based FFT can be replaced with a nonuniform 4D grid-free NFFT. As the difference between the number of sample points in each method grows according to Table 1, even a cascade 4D NDFT over the spherical discretization can be faster, with the additional flexibility it offers in choosing the frequency domain grid size on-the-fly independently of the physical domain sample size. The NDFT does not require the additional step of bump function rasterization over the uniform grid, which is basically a cascade computation of the convolution of the knots and the conical kernel in (13). It rather incorporates that step as a pointwise multiplication with the kernel's frequency domain representation in (24) which can be precomputed to full precision. The comparison is shown in Fig. 8 for different number of modes m' over the 4D frequency grid, which demonstrates an advantage to NDFT for $m > 2^{15} = 32768$ and $m' < 2^{16} = 65536$ on the CPU, and for $m > 2^{18} = 262144$ and $m' < 2^{12} = 4096$ on the GPU. This can be further improved using the optimal NFFT [10]. Unfortunately, NFFTs have been implemented on the GPU [24] for 1, 2, and 3D, while at present the 4D NFFT is available only on the CPU [38].

Lastly, we test the performance for time-critical computation of the correlation predicate for a single configuration, using the method presented in Section 4.1 based on [22] via truncated frequency grid integration in (50). Fig. 9 shows sequential integration time on the CPU for different choices of the number of retained modes $m' \leq m$. An almost linear speed-up of m/m' (as expected from the theory) is achieved, and the collision predicate is computed in less than a millisecond for $m' < 2^{12} = 4096$. This enables fast physically-based modeling and multibody dynamics simulations in real-time applications that require a refresh rate

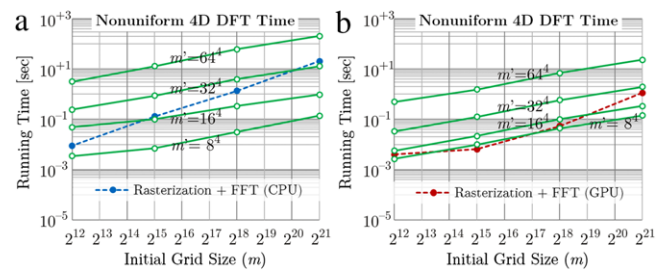


Fig. 8. A comparison of (1) bump function rasterization (i.e., cascade convolution of knots with conical kernels) + 3D FFT; and (2) 4D NDFT + pointwise multiplication with kernel's frequency domain representation, for different number of needed frequencies $m' \leq m$.

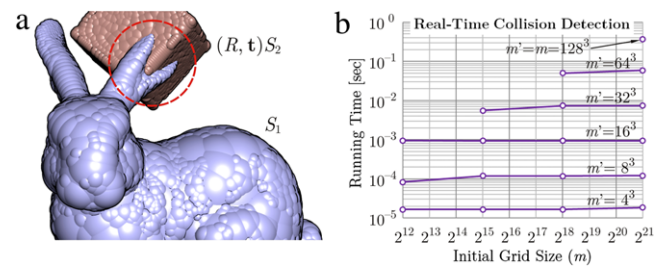


Fig. 9. Time-critical collision detection for a single configuration query by integrating over the $m' \leq m$ dominant modes in the frequency domain. The sub-millisecond region is considered 'real-time'.

of 1 kHz for graphics and haptics feedback [16]. An important research question concerns the development of a *hybrid* method that further improves the performance by using this method alongside a sphere-tree traversal used in [13–16], and limits the integration over the NDFT of fewer primitives at the tree leaves.

6. Conclusion

Analytic methods for geometric modeling have shown great promise in formulating and solving important problems in terms of convolutions, typically implemented by uniform grid-based sampling to enable the use of the efficient FFT. We proposed a versatile analytic paradigm built around a nonuniform

discretization scheme that approximates the shape with a union of 3D balls, equivalent to a nonuniform sample of 4D knots in an elevated space. We showed that this can be conceptualized as a 4D Minkowski sum, analytically represented by a convolution of the knots with a conical kernel. As a result of the spherical symmetry, the discretization structure is closed under rigid motions and Minkowski sums, hence the ball/cone geometry and its defining kernel abstract the continuous geometry away in a consistent manner across configuration space correlations, allowing the numerical algorithms to operate on the discrete set alone.

Among the important applications we referred to collision detection, shape complementarity, and morphological operations, which are central to important problems in motion/path planning, physically-based modeling, manufacturing automation, protein docking, and more. We showed that the spherical discretization allows for an efficient allocation of time and memory resources to capturing the details of geometric features, by rapidly filling the large interior regions of both shape and configuration pointsets. The CPU and GPU implementations were presented, demonstrating speed-ups up to several orders of magnitude.

This research opens up promising theoretical and computational directions for future studies on analytic methods that are emerging in solid and physical modeling.

Acknowledgments

The authors would like to thank Saigopal Nelaturi from Palo Alto Research Center for the helpful discussions and constructive feedback. This work was supported in part by the National Science Foundation grants CMMI-1200089, CMMI-0927105, and CNS-0927105. The responsibility for any errors and omissions lies solely with the authors.

Appendix A. Sampling algorithm

Although the main focus of this article is on how to work with given spherical decompositions in the analytic realm regardless of the method used for their generation [13–16], a ‘good’ spherical sampling algorithm is essential to gain practical advantage over uniform sampling. Here we present the details of our new algorithm used to generate the decomposition in Table 1 and Figs. 5 and 6 of Section 5, along with its topological properties and geometric error bounds (Propositions 1–3). We compare our algorithm, in terms of these qualitative properties as well as arithmetic complexity, to the sphere-packing algorithm by Weller and Zachmann [16] (hereon abbreviated as W&Z), which is considered state-of-the-art for collision detection and proximity queries, and is successfully applied to real-time physically-based modeling and virtual reality (graphics/haptics) [16].

The following lemma from [22] will be central to put the approximation error bounds into perspective:

Lemma 3. Given arbitrary sets $X_1, X_2 \subset \mathbb{R}^3$ and $\epsilon \geq 0$,

$$\left. \begin{aligned} X_1 &\subseteq X_2 \oplus B(\mathbf{0}, \epsilon) \\ X_2 &\subseteq X_1 \oplus B(\mathbf{0}, \epsilon) \end{aligned} \right\} = d_H(X_1, X_2),$$

where $d_H(X_1, X_2)$ is the symmetrical (i.e., the maximum of left and right) Hausdorff L^2 -metric.¹⁴

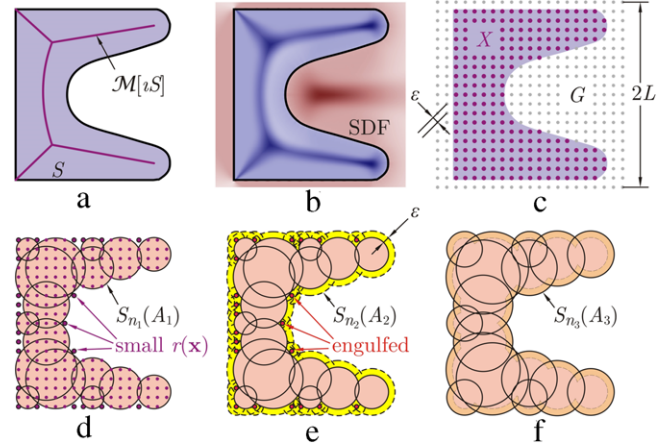


Fig. A.10. The various geometric constructs in Algorithm 1. It precomputes SDF on a grid G (a)–(c), then generates balls centered at grid nodes $X = G \cap S$ in descending SDF order to obtain A_1 (d), expands the balls by ϵ to obtain $A_2 = A_1 + (\mathbf{0}, \epsilon)$ (e), and cleans up the engulfed balls to obtain $A_3 \subset A_2$ (f).

To approximate an arbitrary r -set $S \in \mathcal{S}$ with a finite union of n (possibly overlapping) nonequiradius spherical balls $S_n(A) = \bigcup_{1 \leq i \leq n} B(\mathbf{x}_i, r_i)$, where the balls are encoded by $A = \{(\mathbf{x}_i, r_i)\}_{1 \leq i \leq n} \subset \mathbb{R}^4$, we use a greedy algorithm similar to W&Z [16]. One of the main differences is that W&Z uses the distance field as the greedy criterion, while our algorithm employs the SDF [34] to create superior decompositions similar to the outcomes of MA-based methods [14,15] without the need to explicitly compute the numerically unstable MA/MAT. The SDF is a real-valued function that can be thought of as a ‘dissipated’ continuous extension of the indicator function of the MA (Fig. A.10(b)), whose ridges constitute the interior regions with extensive approximate nearest neighbors on the boundary ∂S , and is computed directly from the distance field [34].

Step 0: Precomputations. First, the algorithm precomputes the distance function and SDF over a uniform grid of m nodes sampled to cover the r -set’s bounding volume. Without loss of generality, we assume a bounding box of edge length $2L > 0$ and a grid resolution of $2L/m^{1/3}$. Let $G = (2L/m^{1/3})\mathbb{Z}^3 \cap [-L, +L]^3$ with $|G| = m$ be the collection of grid nodes, and $X = G \cap S$ be the subset of the nodes that encodes a bitmap approximation of the r -set. The sample X is said to be an ϵ -covering of S where $\epsilon = \sqrt{3}(L/m^{1/3})$ is the diagonal half-length of a grid cell, since for every point $\mathbf{x}' \in S$ there exists a node $\mathbf{x} \in X$ such that $\|\mathbf{x} - \mathbf{x}'\|_2 \leq \epsilon$ [4], which implies

Lemma 4. $d_H(X, S) \leq \epsilon$, where $X = G \cap S$.

Proof. By definition $X \subset S \subset S \oplus B(\mathbf{0}, \epsilon)$. Additionally, $\|\mathbf{x} - \mathbf{x}'\|_2 \leq \epsilon$ for all $\mathbf{x} \in X$ and $\mathbf{x}' \in S$, hence $S \subset X \oplus B(\mathbf{0}, \epsilon)$. Applying Lemma 3 yields the result. The equality holds (i.e., $d_H(X, S) = \epsilon$) unless none of the grid cell centers in between the nodes belongs to S . \square

We will show that the same error bound holds for the spherical decomposition, which forms a basis of comparison in Section 5 between the uniform sample and spherical sample, in terms of their deviation from the original solid measured by ϵ (or equivalently, m).

Step 1: Decomposition. The algorithm sorts X with respect to the SDF values and arranges the nodes into a priority queue, followed by n rounds of selection. At each round, the grid node $\mathbf{x} \in X$ with maximal SDF is selected, and a ball $B(\mathbf{x}, r(\mathbf{x}))$ is generated, i.e., the 4-tuple $(\mathbf{x}, r(\mathbf{x}))$ is added to the (initially empty) output set A_1 , where $r(\mathbf{x}) = d(\mathbf{x}, \partial S) = \min_{\mathbf{x}' \in \partial S} \|\mathbf{x} - \mathbf{x}'\|_2$ is the distance function. A subset of the grid points inside this new ball

¹⁴ $d_H(X_1, X_2) = \max \{ \sup_{\mathbf{x} \in X_1} d(\mathbf{x}, X_2), \sup_{\mathbf{x} \in X_2} d(\mathbf{x}, X_1) \}$.

Table A.2

Arithmetic complexities of spherical samples generated by our SDF-based algorithm (B&I) using different protrusion factors, compared to those of W&Z [16].

Initial grid		B&I, $\mu = 0.25$		B&I, $\mu = 0.50$		B&I, $\mu = 1.00$		W&Z
m	ϵ	$n_{1,2}$	n_3	$n_{1,2}$	n_3	$n_{1,2}$	n_3	n
2^{12}	0.0541	318	36	307	53	275	67	334
2^{15}	0.0271	1,552	161	1,445	251	1,295	324	1,024
2^{18}	0.0135	6,842	1024	6,374	1048	5,436	1311	15,592
2^{21}	0.0068	26,217	3686	23,891	4422	20,674	5160	89,030

Algorithm 1: The SDF-based spherical sampling algorithm
 B&I : $(S, G; L, \mu) \rightarrow (A_1, A_2, A_3)$.

Input: $S \in \mathcal{S}$ (represented with any scheme that supports distance and inclusion queries), $G \subset \mathbb{R}^3$, $L > 0$, and $0 \leq \mu \leq 1$;

Output: $A_1, A_2, A_3 \subset \mathbb{R}^4$;

Step 0: Precomputations:

Constant $\epsilon \leftarrow \sqrt{3}(L/|G|)^{\frac{1}{3}}$;

Using the inclusion test, separate $X = G \cap S$;

Compute the distance function over all X ;

Compute and sort[†] the SDF [34] over all X ;

Step 1: Decomposition:

Initialize $Y \leftarrow \emptyset$; $A_1 \leftarrow \emptyset$;

while $|Y| < |X|$ (implies $Y \subset X$ but $Y \neq X$) **do**

 Select[†] $\mathbf{x} \in X$ with the maximal SDF;

 Modify $A_1 \leftarrow (A_1 \cup \{\mathbf{x}, r(\mathbf{x})\})$; //sampled

for $\mathbf{x}' \in X$ **do**

if $\mathbf{x}' \in B(\mathbf{x}, r(\mathbf{x}))$ **then**

 Modify* $Y \leftarrow (Y \cup \{\mathbf{x}'\})$; //covered

if $\|\mathbf{x} - \mathbf{x}'\|_2 - |r(\mathbf{x}) - r(\mathbf{x}')| \leq \mu r(\mathbf{x}')$ **then**

 Modify[†] SDF[\mathbf{x}''] $\leftarrow -\infty$; //popped

Step 2: Expansion:

Initialize $A_2 \leftarrow \emptyset$;

for $(\mathbf{x}_1, r_1) \in A_1$ **do**

$A_2 \leftarrow (A_2 \cup \{(\mathbf{x}_1, r_1 + \epsilon)\})$; //expanded

Step 3: Reduction:

Initialize $A_3 \leftarrow A_2$;

for $(\mathbf{x}_1, r_1) \in A_1$ **do**

for $(\mathbf{x}_2, r_2) \in A_2$ **do**

if $\|\mathbf{x}_2 - \mathbf{x}_1\|_2 \leq (r_2 - r_1)$ **then**

$A_3 \leftarrow (A_3 - \{(\mathbf{x}_1, r_1 + \epsilon)\})$; //popped

break for;

//Implemented via a priority queue ([†]) and a binary array (*).

are identified with the following condition, and eliminated from the SDF-sorted queue for the next rounds:

$$Q(\mathbf{x}; \mu) = \{\mathbf{x}' \in X \mid \|\mathbf{x} - \mathbf{x}'\|_2 - |r(\mathbf{x}) - r(\mathbf{x}')| \leq \mu r(\mathbf{x}')\}. \quad (\text{A.1})$$

The left-hand side of the inequality gives a measure of how $B(\mathbf{x}', r(\mathbf{x}'))$ protrudes outside $B(\mathbf{x}, r(\mathbf{x}))$, which is normalized by the smaller radius $r(\mathbf{x}')$ and compared to the ‘protrusion factor’ $0 \leq \mu \leq 1$. On the one end, $Q(\mathbf{x}; 0)$ includes only the grid nodes $\mathbf{x}' \in X \cap B(\mathbf{x}, r(\mathbf{x}))$ with $B(\mathbf{x}', r(\mathbf{x}')) \subset B(\mathbf{x}, r(\mathbf{x}))$, i.e., nodes that would contribute redundant balls already covered. On the other end, $Q(\mathbf{x}; 1)$ includes the entire $X \cap B(\mathbf{x}, r(\mathbf{x}))$ and eliminates all covered points from the future rounds. The former gives a more conservative coverage and creates better approximations to the original shape in terms of topology (e.g., fewer void spaces) and geometry (e.g., similar surface curvature) in practice. However, it generates a larger number of balls than the latter – especially over the thinner features of the shape – thus takes longer to finish. More importantly, a too small choice (e.g., $\mu < 0.1$) might defeat the advantage of spherical sampling with $|A_1| = n_1$ over uniform

sampling with $|X| = O(m)$ reliant on the assumption $n_1 \ll m$. Our experiments suggest a choice of $\mu = 0.25\text{--}0.30$ to be a good trade-off (see Table A.2).

At each round of the algorithm, all $\mathbf{x}' \in B(\mathbf{x}, r(\mathbf{x}))$ are marked by Boolean flags that indicate their containment in at least one ball. The algorithm repeats this process until all points in the uniform sample X are covered by the balls, i.e., until $Y = X$ where $Y = G \cap S_{n_1}(A_1)$ is the set of grid nodes covered so far.

Proposition 1. $d_H(S_{n_1}(A_1), S) \leq \epsilon$.

Proof. At the end of step 1, $X \subset S_{n_1}(A_1)$ hence for every point $\mathbf{x}' \in S$, $d(\mathbf{x}', S_{n_1}(A_1)) \leq d(\mathbf{x}', X)$, and since $d(\mathbf{x}', X) \leq \epsilon$, $S \subset S_{n_1}(A_1) \oplus B(\mathbf{0}, \epsilon)$. On the other hand, choosing the minimum distance to the boundary as ball radii guarantees $S_{n_1}(A_1) \subset S \subset S \oplus B(\mathbf{0}, \epsilon)$. Applying Lemma 3 yields the result. \square

It is worthwhile noting that the above proof only makes use of $G \subset S_{n_1}(A_1) \subset S$, hence Proposition 1 is valid for the W&Z algorithm [16] as well, providing a basis of comparison between the two (see Table A.2).

An inevitable difficulty with this approach, regardless of the greedy criterion (e.g., distance-based [16] or SDF-based) are topological discrepancies between the solid and its spherical approximation due to the possible void spaces (of feature sizes no larger than 2ϵ) left in between covered grid nodes. Although letting $m \gg 1$ and $\mu \ll 1$ can alleviate the problem, it does not necessarily eliminate it especially near the boundary. In addition to the trapped internal cavities, the algorithm generates numerous small disconnected balls near the boundary where $r(\mathbf{x}) < \epsilon$ as it approaches the final rounds. The next two steps are meant to solve these problems.

Step 2: Expansion. The previous step generates a decomposition that is strictly contained in the original solid, which is unnecessary for our purposes. In the next step, the algorithm expands all the balls by increasing their radii with $\epsilon = \sqrt{3}(L/m)^{\frac{1}{3}}$, which gives

$$A_2 = \{\mathbf{a}_2 = (\mathbf{x}_1, r_1 + \epsilon) \mid \mathbf{a}_1 = (\mathbf{x}_1, r_1) \in A_1\}, \quad (\text{A.2})$$

i.e., $S_{n_2}(A_2) = S_{n_1}(A_1) \oplus B(\mathbf{0}, \epsilon)$. It is easy to show that $S_{n_2}(A_2)$ is in fact as good of an approximation as $S_{n_1}(A_1)$ in terms of Hausdorff error measure.

Proposition 2. $d_H(S_{n_2}(A_2), S) \leq \epsilon$.

Proof. On the one hand, $S_{n_1}(A_1) \subset S \subset S \oplus B(\mathbf{0}, \epsilon)$. On the other hand, $S \subset S_{n_1}(A_1) \oplus B(\mathbf{0}, \epsilon) \subset S_{n_2}(A_2) \oplus B(\mathbf{0}, \epsilon)$. Applying Lemma 3 yields the result. \square

Moreover, the ϵ -expansion ‘repairs’ the inter-cellular cavities and disconnected balls, noting that the distance between every pair of balls corresponding to A_1 is upperbounded by 2ϵ . Therefore, although the algorithm cannot guarantee topological equivalence (i.e., homeomorphism) between S and any decomposition, simply because it only relies on the incomplete shape representation provided by the discrete set X – which fails to capture topological information of the features that are smaller than the sampling resolution – $S_{n_2}(A_2)$ does guarantee a weaker equivalence, i.e.,

homeomorphism with a *voxelization* implied by the uniform sample X , while $S_{n_1}(A_1)$ does not.

Step 3: Reduction. Although $S_{n_2}(A_2)$ exhibits relatively more desirable topological properties than $S_{n_1}(A_1)$, they both have the same complexity $n_1 = n_2$. The former satisfies ‘conservative coverage’ ($S_{n_2}(A_2) \supset S$) while the latter fulfills ‘strict containment’ ($S_{n_1}(A_1) \subset S$). The main purpose of the expansion, however, was to eliminate the small balls at the grid nodes near the surface with $r(\mathbf{x}) < \epsilon$ that could not be covered by any other ball in A_1 . These artifacts can comprise a significant fraction of all balls, adding to the numerical complexity with little contribution to the volume, while they adversely affect the quality of discretization by introducing high local curvatures. The question is whether we can obtain a third discretization in between the two, i.e., a sample A_3 such that

$$S_{n_1}(A_1) \subset S_{n_3}(A_3) \subset S_{n_2}(A_2), \quad (\text{A.3})$$

which retains the desired topological properties, eliminates the undesired small balls, guarantees the same error bounds, and is smaller in size ($|A_3| = n_3 < n_{1,2}$).

The next step of the algorithm reduces A_2 into A_3 by eliminating the sample points whose corresponding original (i.e., not expanded) balls encoded by $(\mathbf{x}_1, r_1) \in A_1$ were contained in some other expanded ball corresponding to $(\mathbf{x}'_1, r'_1 + \epsilon) \in A_2$, $(\mathbf{x}_1, r_1) \neq (\mathbf{x}'_1, r'_1)$:

$$A_3 = \{\mathbf{a}_2 = (\mathbf{x}_1, r_1 + \epsilon) \mid \mathbf{a}_1 = (\mathbf{x}_1, r_1) \in A_1,$$

$$B(\mathbf{x}_1, r_1) \not\subset S_{n_2-1}(A_2 - \{\mathbf{a}_2\})\}. \quad (\text{A.4})$$

A significant number of small balls in A_1 near the surface are thus ‘engulfed’ by the expanded neighbor balls in A_2 – e.g., constituting %75–%85 of all balls for the Stanford Bunny in Fig. A.11 with $\mu = 0.25$ –1.00, which grows by decreasing μ . Although the new decomposition neither contains nor is contained in the original solid, the approximation error bound still holds.

Proposition 3. $d_H(S_{n_3}(A_3), S) \leq \epsilon$.

Proof. By definitions in (A.2) and (A.4), $A_3 \subset A_2$ hence $S_{n_3}(A_3) \subset S_{n_2}(A_2) \subset S \oplus B(\mathbf{0}, \epsilon)$. On the other hand, every ball encoded by $(\mathbf{x}_1, r_1) \in A_1$ is contained in at least one ball corresponding to $(\mathbf{x}_2, r_2 + \epsilon) \in A_2$. If this is realized only by $(\mathbf{x}_2, r_2) := (\mathbf{x}_1, r_1)$, noting that always $B(\mathbf{x}_1, r_1) \subset B(\mathbf{x}_1, r_1 + \epsilon)$, then its ϵ -expansion is included in A_3 by definition in (A.4). Otherwise, there exists at least another ball represented by $(\mathbf{x}_2, r_2) \neq (\mathbf{x}_1, r_1)$ whose ϵ -expansion is included in A_3 , i.e., $(\mathbf{x}_2, r_2 + \epsilon) \in A_3$, and $B(\mathbf{x}_1, r_1) \subset B(\mathbf{x}_2, r_2 + \epsilon)$. In either case, $B(\mathbf{x}_1, r_1)$ is included in $S_{n_3}(A_3)$, hence $S_{n_1}(A_1) \subset S_{n_3}(A_3)$. Therefore, $S \subset S_{n_1}(A_1) \oplus B(\mathbf{0}, \epsilon)$ implies $S \subset S_{n_3}(A_3) \oplus B(\mathbf{0}, \epsilon)$. Applying Lemma 3 yields the result. \square

All aforementioned steps are summarized in Algorithm 1. Fig. A.10 illustrates the different geometric objects involved in the algorithm, including the output decompositions of each step. Once the decomposition is generated, a variety of methods can be used to create sphere-tree hierarchies—e.g., the simple bottom-up merging algorithm in [14] that groups and merges a pre-specified number of balls that are near each other in A_3 (i.e., the children) into their smallest enclosing ball (i.e., the parent).

Table A.2 compares the numerical complexity of each decomposition $|A_1| = |A_2|$, and $|A_3|$ with that of W&Z [16] for the Stanford Bunny in Fig. 5(a) using different grid resolutions and protrusion factors. We observe that unlike the case for $n_1 = n_2$, n_3 decreases with more conservative choices of μ , making the smaller value of $\mu = 0.25$ even more favorable. The same value was used to generate the results in Table 1 and Figs. 5 and 6 of Section 5. After this reduction, the output of our algorithm contains fewer balls than W&Z [16] by an order of magnitude, allowing faster termination—in a few minutes for the slowest case. Fig. A.11 illustrates $S_{n_3}(A_3)$ corresponding to the parameters in Table A.2.

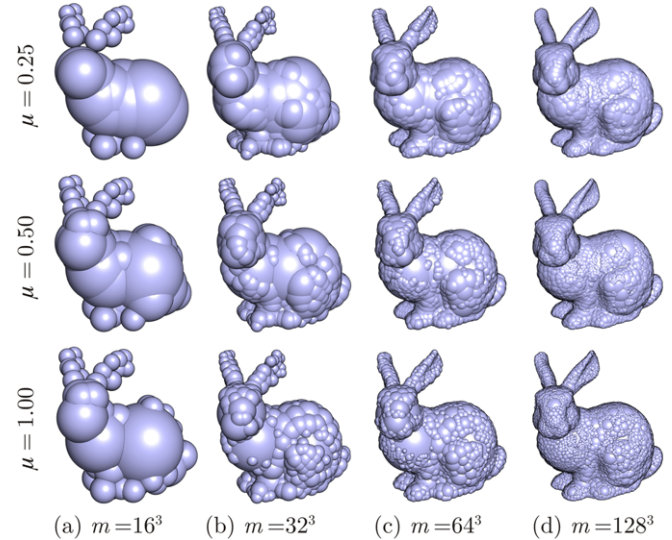


Fig. A.11. The SDF-based spherical decomposition $S_{n_3}(A_3)$ for different initial grid sizes and protrusion factors in Table A.2.

Appendix B. The motion group

According to Requicha [26], a ‘rigid body’ $[S]$ is an equivalence class of congruent r -sets related by the elements in $\mathcal{C} := SE(3)$, i.e., $[S] = \{MS \mid M \in SE(3)\} \subset \mathcal{R}$. The special Euclidean group $SE(3) \cong SO(3) \times \mathbb{R}^3$ represents the configuration space of all rigid motions (i.e., distance and orientation preserving affine maps). It extends the special orthogonal group $SO(3)$, which contains all proper rotations (i.e., distance and orientation preserving linear maps) over its action on \mathbb{R}^3 :

$$SO(3) = \{R \in M(3) \mid RR^T = R^T R = I, \det(R) = +1\}, \quad (\text{B.1})$$

with matrix multiplication used as the group operation. $M(3) \cong \mathbb{R}^9$ is the space of all 3×3 matrices, and $I = id_{M(3)}$ is the identity matrix.

As a result of the semidirect product, every rigid motion can be parameterized by a tuple $M = (R, \mathbf{t}) \in SE(3)$ with the following properties:

- The group operation is defined as $(R_1, \mathbf{t}_1)(R_2, \mathbf{t}_2) = (R_1 R_2, \mathbf{t}_1 + R_1 \mathbf{t}_2)$.
- The inversion is defined as $(R, \mathbf{t})^{-1} = (R^T, -R^T \mathbf{t})$.
- The action on \mathbb{R}^3 is defined as $(R, \mathbf{t})\mathbf{x} = (R\mathbf{x}) + \mathbf{t}$ for all $\mathbf{x} \in \mathbb{R}^3$ (i.e., a noncommutative sequence of a rotation $\mathbf{x} \mapsto R\mathbf{x}$ followed by a translation $\mathbf{x} \mapsto \mathbf{x} + \mathbf{t}$), hence $(R, \mathbf{t})^{-1}\mathbf{x} = R^T(\mathbf{x} - \mathbf{t})$.
- Noting that $SE(3)$ is a subgroup of $SE(4)$, we can define the action of $SE(3)$ on 4D geometry simply as $(R, \mathbf{t})(\mathbf{x}, r) := ((R, \mathbf{t})\mathbf{x}, r)$ for $(\mathbf{x}, r) \in \mathbb{R}^4$.

Therefore, every moved instance of S is described by $(R, \mathbf{t})S := \{(R, \mathbf{t})\mathbf{x} \mid \mathbf{x} \in S\} \in [S]$.

Since $SE(3)$ is a Riemannian manifold, any pair of configurations can be joined along a curve that minimizes the geodesic distance, defined as the Riemannian metric

$$d(M_1, M_2) := \left(\|\ln(R_1^T R_2)\|_2^2 + \|\mathbf{t}_2 - \mathbf{t}_1\|_2^2 \right)^{\frac{1}{2}}, \quad (\text{B.2})$$

where the logarithmic term can be viewed as the geodesic $d(R_1, R_2)$ over the manifold $SO(3)$, which is in a 1:2 homeomorphism with a unit 3-sphere embedded in \mathbb{R}^4 , $\ln(\cdot)$ is the matrix logarithm, and $\|\cdot\|_2$ is the Frobenius norm, which is the same as the Euclidean L^2 -norm of a 9-vector formed from the elements of a 3×3 matrix [4]. Using (B.2), $SE(3)$ also forms a metric (hence a topological) space.

Appendix C. Fourier transforms

A function $f : \mathbb{R}^3 \rightarrow \mathbb{C}$ is square-integrable (i.e., $f \in L^2(\mathbb{R}^3)$) if $v(|f|^2) = v(ff)$ defined by the integral in (4) converges, i.e., if its L^2 -norm $\|f\|_2 := \sqrt{\langle f, f \rangle}$ exists, where the function inner product $\langle \cdot, \cdot \rangle : L^2(\mathbb{R}^3) \times L^2(\mathbb{R}^3) \rightarrow \mathbb{C}$ is

$$\langle f_1, f_2 \rangle = \int_{\mathbb{R}^3} f_1(\mathbf{x}) \bar{f}_2(\mathbf{x}) d\mathbf{x} \quad (\text{C.1})$$

in which $\bar{f}(\mathbf{x}) = \overline{f(\mathbf{x})}$ denotes conjugation. Also related are the cross-correlation and convolution operators $\star, * : L^2(\mathbb{R}^3) \times L^2(\mathbb{R}^3) \rightarrow L^2(\mathbb{R}^3)$:

$$(f_1 \star f_2)(\mathbf{t}) = \int_{\mathbb{R}^3} \bar{f}_1(\mathbf{x}) f_2(\mathbf{x} + \mathbf{t}) d\mathbf{x}, \quad (\text{C.2})$$

$$(f_1 * f_2)(\mathbf{t}) = \int_{\mathbb{R}^3} f_1(\mathbf{x}) \bar{f}_2(\mathbf{x} - \mathbf{t}) d\mathbf{x}, \quad (\text{C.3})$$

in which $\tilde{f}(\mathbf{x}) = f(-\mathbf{x})$ denotes reflection. It can be verified by a simple change of variables that $\langle f_1, (\tilde{f}_2 \circ T^{-1}) \rangle = \langle \tilde{f}_2 \star f_1 \rangle(\mathbf{t}) = \langle f_1 * \tilde{f}_2 \rangle(\mathbf{t})$ where $T^{-1}\mathbf{x} = \mathbf{x} - \mathbf{t}$ is a shift.

The continuous Fourier transform (CFT) often denoted by $\mathcal{F} : L^2(\mathbb{R}^3) \rightarrow L^2(\mathbb{R}^3)$ is defined as

$$\mathcal{F}\{f\}(\boldsymbol{\omega}) = \int_{\mathbb{R}^3} f(\mathbf{x}) e^{-2\pi i(\boldsymbol{\omega} \cdot \mathbf{x})} d\mathbf{x}, \quad (\text{C.4})$$

which can be conceptualized as a decomposition of the L^2 -function along orthogonal basis $e^{2\pi i(\boldsymbol{\omega} \cdot \mathbf{x})}$ (see (21)). Using the notations $\hat{f} = \mathcal{F}\{f\}$ and $f = \mathcal{F}^{-1}\{\hat{f}\}$, the following properties are easily verified:

- Translation in the physical domain converts to a multiplier in the frequency domain, i.e.,

$$\mathcal{F}\{f \circ T^{-1}\} = \mathcal{F}\{\zeta_{\mathbf{t}}\} \mathcal{F}\{f\} = \hat{\zeta}_{\mathbf{t}} \hat{f}, \quad (\text{C.5})$$

where $\zeta_{\mathbf{t}}(\mathbf{x}) = (\delta^3 \circ T^{-1})(\mathbf{x}) = \delta^3(\mathbf{x} - \mathbf{t})$ whose CFT is the sinusoidal basis function $\hat{\zeta}_{\mathbf{t}}(\boldsymbol{\omega}) = e^{-2\pi i(\boldsymbol{\omega} \cdot \mathbf{t})}$.

- As a result of the linearity of (C.4), linear maps (e.g., reflections and rotations) are preserved under the CFT, e.g., $\hat{\hat{f}} = \hat{f}$ and

$$\mathcal{F}\{f \circ R^T\} = \mathcal{F}\{f\} \circ R^T = \hat{f} \circ R^T, \quad (\text{C.6})$$

- If $f = \bar{f}$ (i.e., f is real-valued), one obtains $\hat{\hat{f}} = \bar{\hat{f}}$, i.e., the CFT converts reflection to conjugation. Substituting from the previous property results in $\hat{\hat{f}} = \bar{\hat{f}}$, a property known as the *Hermitian symmetry*.

- The inner product structure is preserved under the CFT, i.e., $\langle f_1, f_2 \rangle = \langle \hat{f}_1, \hat{f}_2 \rangle$, an important result known as *Parseval's theorem*.

- The convolution in physical domain converts to pointwise multiplication in the Fourier domain, i.e.,

$$\mathcal{F}\{f_1 * f_2\} = \mathcal{F}\{f_1\} \mathcal{F}\{f_2\} = \hat{f}_1 \hat{f}_2, \quad (\text{C.7})$$

a crucial result known as the *convolution theorem*.

By discretizing the function's domain into a finite set of knots, the integral in (C.4) can be approximated by a Riemann sum, which can be expressed in terms of the discrete Fourier transform (DFT):

$$\hat{f}_j = \sum_{i=1}^n f_i e^{-2\pi i(\boldsymbol{\omega}_j \cdot \mathbf{x}_i)}, \quad (1 \leq j \leq n). \quad (\text{C.8})$$

If both physical and frequency sample points are equispaced, (C.8) can be computed for all sample points in $O(n \log n)$ time (in contrast to $O(n^2)$ time for cascade computation) using the radix-2 fast Fourier transform (FFT) [9]. If either (or both) of the samples

are nonequispaced, the so-call nonequispaced FFT (NFFT) [10] can be used, which takes $O((\alpha n) \log(\alpha n))$ time with an 'over-sampling factor' of $\alpha = O(1)$. The latter is advantageous over the former if the nonuniform sampling is sparse.

On a final note, the extension of the aforementioned 3D definitions to 4D is trivial. The reader is referred to [39] for an in-depth study of harmonic analysis.

References

- [1] Roerdink JBTM. Group morphology. *Pattern Recognit* 2000;33(6):877–95.
- [2] Lysenko M, Shapiro V, Nelaturi S. Non-commutative morphology: Shapes, filters, and convolutions. *Comput Aided Geom Design* 2011;28(8):497–522.
- [3] Lozano-Perez T. Spatial planning: A configuration space approach. *IEEE Trans Comput* 1983;C-32(2):108–20.
- [4] Nelaturi S, Shapiro V. Configuration products and quotients in geometric modeling. *Comput Aided Des (CAD)* 2011;43(7):781–94.
- [5] Behandish M, Ilieş HT. Peg-in-hole revisited: A generic force model for haptic assembly. *J Comput Inf Sci Eng (JCISE)* 2015; <http://dx.doi.org/10.1115/1.4030749> [in press].
- [6] Behandish M, Ilieş HT. Haptic assembly using skeletal densities and Fourier transforms. *J Comput Inf Sci Eng (JCISE)* 2015 [in press].
- [7] Bajaj CL, Chowdhury R, Siddahanavalli V. F2Dock: Fast Fourier protein–protein docking. *IEEE/ACM Trans Comput Biol Bioinf (TCBB)* 2011;8(1):45–58.
- [8] Chernov N, Stoyan Y, Romanova T. Mathematical model and efficient algorithms for object packing problem. *Comput Geom* 2010;43(5):535–53.
- [9] Cooley JW, Tukey JW. An algorithm for the machine calculation of complex Fourier series. *Math Comp* 1965;19(90):297–301.
- [10] Potts D, Steidl G, Tasche M. Fast Fourier transforms for nonequispaced data: A tutorial. In: *Modern sampling theory. Applied and numerical harmonic analysis*. Springer-Verlag; 2001. p. 247–70.
- [11] Sagardia M, Stouraitis T, Silva JL. A new fast and robust collision detection and force computation algorithm applied to the physics engine Bullet: Method, integration, and evaluation. In: 2014 conference and exhibition of the european association of virtual and augmented reality. EuroVR'2014. 2014.
- [12] Gottschalk S, Lin MC, Manocha D. OBBTree: A hierarchical structure for rapid interference detection. In: *Proceedings of the 23rd annual conference on computer graphics and interactive techniques. SIGGRAPH'1996*. New York, NY, USA. 1996. p. 171–80.
- [13] O'Sullivan C, Dingliana J. Real-time collision detection and response using sphere-trees. In: *Spring conference on computer graphics*. Bratislava, Slovakia. 1999. p. 83–92.
- [14] Hubbard PM. Approximating polyhedra with spheres for time-critical collision detection. *ACM Trans Graph (TOG)* 1996;15(3):179–210.
- [15] Bradshaw G, O'Sullivan C. Adaptive medial-axis approximation for sphere-tree construction. *ACM Trans Graph (TOG)* 2004;23(1):1–26.
- [16] Weller R, Zachmann G. Inner sphere trees and their application to collision detection. In: Coquillart S, Brunnert G, Welch G, editors. *Virtual realities*. Springer Vienna; 2011. p. 181–202.
- [17] Kavraki LE. Computation of configuration-space obstacles using the fast Fourier transform. *IEEE Trans Robot Autom* 1995;11(3):408–13.
- [18] Chen R, Weng Z. A novel shape complementarity scoring function for protein–protein docking. *Proteins: Struct Funct Bioinform* 2003;51(3):397–408.
- [19] Mirtich B. V-Clip: Fast and robust polyhedral collision detection. *ACM Trans Graph (TOG)* 1998;17(3):177–208.
- [20] Kim CE, Vance JM. Collision detection and part interaction modeling to facilitate immersive virtual assembly methods. *J Comput Inf Sci Eng (JCISE)* 2004;4(2):83–90.
- [21] Jimenez P, Thomas F, Torras C. 3D collision detection: A survey. *Comput Graph* 2001;25(2):269–85.
- [22] Lysenko M. Fourier collision detection. *Int J Robot Res* 2013;32(4):483–503.
- [23] Ritchie DW. Recent progress and future directions in protein–protein docking. *Curr Protein Pept Sci* 2008;9(1):1–15.
- [24] Kunis S, Kunis S. The nonequispaced FFT on graphics processing units. *Proc Appl Math Mech (PAMM)* 2012;12(1):7–10.
- [25] Potts D, Steidl G, Nieslony A. Fast convolution with radial kernels at nonequispaced knots. *Numer Math* 2004;98(2):329–51.
- [26] Requicha AG. Mathematical models of rigid solid objects, Production automation project. Technical Memo. No. 28. University of Rochester. 1980.
- [27] Requicha AG. Mathematical foundations of constructive solid geometry: General topology of closed regular sets, Production automation project. Technical Memo. No. 27. University of Rochester. 1978.
- [28] Chazal F, Soufflet R. Stability and finiteness properties of medial axis and skeleton. *J Dyn Control Syst* 2004;10(2):149–70.
- [29] Comba PG. A procedure for detecting intersections of three-dimensional objects. *J ACM (JACM)* 1968;15(3):354–66.
- [30] Ricci A. A constructive geometry for computer graphics. *Comput J* 1973;16(2):157–60.
- [31] Shapiro V. Semi-analytic geometry with R-functions. *Acta Numer* 2007;16:239–303.

- [32] Requicha AG. Representations of rigid solid objects. Production automation project. Technical Memo. No. 29. University of Rochester. 1980.
- [33] Duncan BS, Olson AJ. Shape analysis of molecular surfaces. *Biopolymers* 1993; 33(2):231–8.
- [34] Behandish M, Ilieş HT. Comparative shape analysis using skeletal density functions. Technical Report No. CDL-TR-15-02. Computational Design Laboratory, University of Connecticut; 2015.
- [35] M Behandish. Spherical decomposition of configuration pointsets: an example. Technical Note CDL-TN-15-04. Computational Design Laboratory, University of Connecticut; 2015.
- [36] Schäling B. *The boost C++ libraries*. 2nd ed. XML Press; 2011.
- [37] Frigo M, Johnson SG. The design and implementation of FFTW3. *Proc. IEEE* 2005;93(2):216–31. Special Issue on Program generation, optimization, and platform adaptation.
- [38] Keiner J, Kunis S, Potts D. Using NFFT 3—a software library for various nonequispaced fast Fourier transforms. *ACM Trans Math Softw (TOMS)* 2009; 36(4):19:1–19:30.
- [39] Katznelson Y. *An introduction to harmonic analysis*. 3rd ed. Cambridge University Press; 2004.