Technische Universität München

Wissenschaftszentrum Weihenstephan für Ernährung, Landnutzung und Umwelt
Lehrstuhl für Systemverfahrenstechnik

Crystal Shape Modeling and Convex Geometry – Analysis of Geometric State Spaces

Alexander Reinhold

Vollständiger Abdruck der von der Fakultät Wissenschaftszentrum Weihenstephan für Ernährung, Landnutzung und Umwelt der Technischen Universität München zur Erlangung des akademischen Grades eines Doktor-Ingenieurs (Dr.-Ing.) genehmigten Dissertation.

Vorsitzender:	Univ.-Prof. Dr. H.-C. Langowski
Prüfer der Dissertation:	1. Univ.-Prof. Dr.-Ing. Heiko Briesen 2. Univ.-Prof. Dr. Marco Mazzotti, ETH Zürich/Schweiz 3. Univ.-Prof. Dr.-Ing. Kai Sundmacher, Otto-von-Guericke-Universität Magdeburg

Die Dissertation wurde am 01.08.2014 bei der Technischen Universität München eingereicht und durch die Fakultät Wissenschaftszentrum Weihenstephan für Ernährung, Landnutzung und Umwelt am 10.04.2015 angenommen.

Contents

Nomenclature

Operators and Special Symbols

$(\cdot)^+$	pseudo inverse
$(\cdot)^T$	transpose
$\|\cdot\|$	Euclidean norm (equation 2.2)
$[\cdot,\cdot]$	closed line segment: $[\mathbf{x},\mathbf{y}] = \{\lambda\mathbf{x} + (1-\lambda)\mathbf{y} \mid \lambda \in \mathbb{R}, 0 \leq \lambda \leq 1\}$
$\langle\cdot,\cdot\rangle$	scalar product (equation 2.1)
$\mathbf{u} \bullet \mathbf{v}$	scalar product (equation 2.1)
$\nabla_{\cdot}$	nabla operator: $\nabla_{\mathbf{x}} = \left(\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \ldots\right)^T$
$\mathbf{0}$	column vector of zeros in appropriate dimension
$\mathbf{1}$	column vector of ones in appropriate dimension
conv B	convex hull of B (see equation 2.5)
dim B	dimension spanned by the vectors in the set B
$\mathbb{E}^n$	Euclidean vector space (n-dimensional, see page 8)
$\mathbf{I}_n$	n-dimensional unity matrix
pos B	positive hull of B (see equation 2.6)
$\mathbb{R}^n$	real vector space (n-dimensional, see page 8)
sup	supremum: the least upper bound of the argument

Greek

γ	some constant, typically indexed and used only locally
$\boldsymbol{\lambda}$	vector of scalar coefficients for $\mathcal{S}$-representation
λ_i	scalar coefficient for $\mathcal{S}$-representation
λ_r	radius of added ball for rounded shapes
μ^{measure}	specific measure, indicated by the superscript
μ^{Bn}	additive roundness based on mean width ($n = 1$), mean width ($n = 2$) or volume ($n = 3$)
φ	polar angle
Π	orthogonal projection (used as operator and index)
ρ	polar distance

Roman

$\mathbf{A}$	matrix of facet normals
$\mathbf{A}_{\mathrm{FV}(k)}$	matrix of facet normals for a facet validity cone
$\mathbf{A}_{\mathrm{S},p}$	matrix of facet normals for a simplex partition
$\mathbf{a}_i$	facet normal (unit vector)
$\mathbf{A}_{\mathrm{U},p}$	matrix of facet normals for a unified partition
$\mathbf{A}_\mathrm{V}$	matrix of facet normals for the cone of valid polytopes
B^n	unit n-ball: $B^n = \{\mathbf{x} \mid \|\mathbf{x}\| \leq 1, \mathbf{x} \in \mathbb{R}^n\}$
C	convex cone for which it holds: $\mathbf{x} \in C$, $\lambda \in \mathbb{R}$, $\lambda > 0 \Rightarrow \lambda\mathbf{x} \in C$

$\mathcal{C}_{\mathrm{FV}(k)}$	facet validity cone
$c_{p,(\cdot)}^{\mathrm{measure}}$	coefficient for measure calculation, where p denotes the unified partition or a-type and $(\cdot)$ is a combination of 1, 2 or 3 indices
$\mathcal{C}_{\mathrm{S},q}$	simplex partition
$\mathcal{C}_{\mathrm{U},\,p}$	unified partition
$\mathcal{C}_{\mathrm{V}}$	validity cone
$F(\cdot,\,\cdot)$	support set for which the arguments are a body and a direction (see equation 2.13)
$\mathbf{g}$	physical growth rate
$\mathcal{G}_{n_{\mathrm{C}},\,n_{\mathrm{A}}}$	Grassmann manifold: describes the set of n_{A}-dimensional subspaces that can be embedded in an n_{C}-dimensional space
$\tilde{\mathbf{g}}$	growth rate that is consistent with validity
$\mathbf{h}$	vector of facet distances
$H(\cdot,\,\cdot)$	hyperplane for which the arguments are the facet normal and the facet distance (see equation 2.7)
$H(\cdot,\,\cdot)$	support plane for which the arguments are a body and a direction (see equation 2.11)
$h(\cdot,\,\cdot)$	support function (see equation 2.9)
$\mathbf{h}_{\mathrm{C}}$	constrained $\mathbf{h}$-vector for $\mathcal{H}_{\mathrm{C}}$-representation
$\mathbf{h}_{\mathrm{c}}$	facet distance vector of a boundary polytope that circumscribes a rounded shape
$h_{\mathrm{C},\,i}$	element of constrained $\mathbf{h}$-vector for $\mathcal{H}_{\mathrm{C}}$-representation
$\mathbf{h}_{\mathrm{C,k}}$	facet distance vector of a kernel polytope in $\mathcal{H}_{\mathrm{C}}$-representation for rounded shapes
$\mathbf{h}_{\mathrm{R}}$	vector for the geometric state space of rounded particles
$\mathbf{h}_{\mathrm{k}}$	facet distance vector of a kernel polytope for rounded shapes
$H^{-}(\cdot,\,\cdot)$	halfspace for which the arguments are the facet normal and the facet distance (see equation 2.7)
$H^{-}(\cdot,\,\cdot)$	supporting halfspace for which the arguments are a body and a direction (see equation 2.12)
h_i	facet distance
I_1	total particle number
I_μ	integral of a particle population respective to a measure name or function μ
$\mathbf{M}$	matrix for mapping between vector spaces
$\mathbf{M}_{\mathrm{C}}$	constraint matrix for $\mathcal{H}_{\mathrm{C}}$-representation
$\mathbf{M}_{\mathbf{h}_{\mathrm{C}}\mapsto\mathbf{h}}$	mapping matrix for $\mathcal{H}_{\mathrm{C}}$-representation
$\mathbf{M}_{\mathbf{h}_{\mathrm{C}}\mapsto\mathbf{h}}$	mapping matrix for $\mathcal{H}_{\mathrm{C}}$-representation
n	dimension of a geometric body or general notation for the dimension of a problem
n_{E}	number of extreme points
n_{C}	number of elements in constrained $\mathbf{h}$-vector for $\mathcal{H}_{\mathrm{C}}$-representation
n_{H}	number of facet normals in $\mathcal{H}$-representation
n_{PS}	total number of simplex partitions
n_{PU}	total number of unified partitions
n_{S}	number of structuring elements for $\mathcal{S}$-representation
$\tilde{n}$	number distribution density
P	polytope
$P_{\mathbf{A}}(\mathbf{h})$, $P(\mathbf{h})$	polytope in $\mathcal{H}$-representation
S^{n-1}	unit $(n-1)$-sphere (boundary of n-ball): $S^{n-1} = \{\mathbf{x} \mid \|\mathbf{x}\| = 1, \mathbf{x} \in \mathbb{R}^n\}$
S_i	structuring element (see section 2.2.4)
$\widetilde{V}$	mixed volume (see section 2.3.2)
$\widetilde{w}$	probability distribution density
$\mathbf{x}$	point in Euclidean or real vector space
$\mathbf{x}_{\mathrm{E}}$	extreme point of a polytope
$\mathbf{y}$	point in Euclidean or real vector space

1

Introduction

1.1 Motivation

Particle Shape. Particle shape affects the processing of powders or dispersions and influences various product properties as it is discussed by Briesen [2008] or Borchert [2012]. The shape and shape distribution influences the filterability, dryability, washability and general solids handling during downstream processing. For the final product, the bioavailability, catalyst efficiency, coating behavior or the efficiency of an active pharmaceutical ingredient are affected by the shape of the respective particles. Because of the purity of crystalline products, crystallization processes are particularly important for pharmaceuticals [Variankaval and Cote, 2008]. The above properties are typically linked to crystals dispersed in fluids. In contrast, Adair and Suvaci [2000] indicate the importance of particle shape for composite materials where particles with anisotropic properties like crystals are embedded in either an anisotropic microstructure or an isotropic matrix.

State Spaces. Computing the geometrical properties of a crystal (*e.g.* volume or specific facet areas) is straightforward, once the crystal shape is measured or defined. Corresponding algorithms (*e.g.* volume computation of polytopes) are also efficient enough to simulate the shape evolution of a single crystal while the complexity increases significantly when the disappearance of faces must be detected [Zhang et al., 2006]. However, typical crystalline products are powders or dispersions. For the simulation of these crystal populations, a speedup of computations is inevitable. To achieve this, the viewpoint on a single crystal shape must be lifted to a broader viewpoint that overlooks all possible crystal shapes of a certain crystal system. This new viewpoint looks on a geometrical state space in which each point equals a specific crystal shape. Analyzing this space would aim at revealing the subdomains in which certain crystal faces are present or finding a shortcut to compute the particle volume. At the beginning of this work, this analysis was in it's infancy so that parallel developments are discussed later in section 1.4.

Joy. An alternative motivation for this work would be analogue to Boerrigter [2003] whose introduction is mostly driven by the fascination for crystals and, therefore, includes many references to the early scientific interest in this field. Improving the understanding for the nature of things that are surrounding us (*e.g.* crystals) and discovering new relations between different aspects (*e.g.* crystal structure, growth rates, shape

and connected mathematical concepts) is a motivating prospect by itself. From my personal view, prior arguments to improve industrial processes and product properties are (just) an economical driving force and a mere justification to follow a natural impulse of humankind: enjoying to be curious and ambitious.

1.2 State of the Art

Crystal shape representation. The commonly accepted crystal shape representation is based on a finite selection of faces that can possibly appear during a crystallization process. Since crystals always exhibit certain symmetry, these faces come in groups called crystal forms. Figure 1.1 shows some examples for which the faces of such groups are colored in equal shades. The shape and size of a crystal is then determined by the distance of these faces from the origin of the coordinate system. Because of the symmetry, only a few representative face distances must be considered for a sufficient representation. These face distances constitute the geometric state space that is analyzed in this work.

The current state of the art for the modeling of growing crystal populations with the aforementioned geometric detail is indicated by the dissertation of Christian Borchert [2012]. Before that, similar analysis was only performed for the evolution of single crystals, for crystal populations with constant or similar shape, or for examples with significant simplifications. The associated array of scientific work is briefly reviewed in the last part of the introduction. The remainder of this subsection contains an outline of the present achievements and corresponding open questions.

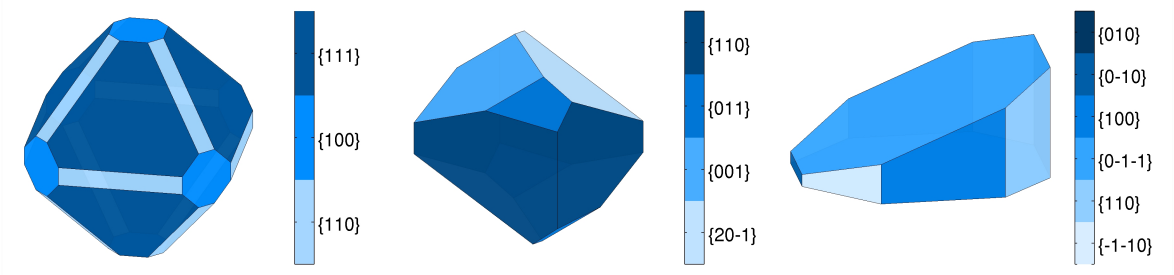

Figure 1.1.: potash alum (left), paracetamol (middle) and lactose (right)

Validity. One of the most imminent issues for crystal shape modeling shall be outlined in the following. Imagine the potash alum crystal in figure 1.1 being altered by increasing the distance for the light blue colored $\{110\}$ faces. These faces will quickly grow out. Next, imagine the face distances increasing further. The obtained combination of face distances seems strange because increasing the distances of the $\{110\}$ faces does not change the shape any further. This problem is denoted by 'validity' throughout this present work and Borchert successfully prevents such invalid combinations of face distances in presented simulations. However, he did not provide a method to handle such invalid combinations with an equal efficiency compared to valid combinations of face distances. This capability is useful to conclude schemes for model reduction and shape approximation that are demonstrated in this work.

Generality. The geometric state space for the potash alum example in figure 1.1 only requires 3 representative face distances that are $h_{\mathrm{C},1}$ for the $\{111\}$ faces, $h_{\mathrm{C},2}$ for the

$\{100\}$ faces and $h_{\mathrm{C},3}$ for the $\{110\}$ faces. The complete set of face distances, expressed by the 26-dimensional vector $\mathbf{h}$, can then be obtained by a mapping: $\mathbf{h} = \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}\mathbf{h}_\mathrm{C}$ from the 3-dimensional vector $\mathbf{h}_\mathrm{C}$. This mapping transfers the symmetry information and is commonly used for the definition of the crystal geometry. However, it is not clear whether arbitrary mapping matrices $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}$ can be chosen or if yet undefined conditions must hold for the framework proposed by Borchert. It turns out that such properties exist so that proper representations that fulfill them are distinguished from improper representations. Fortunately, representations that are constructed according to crystal symmetry are always proper. However, it also follows that it is not possible to treat arbitrary and possibly improper shape representations with the current state of the art. Again, applications are model reduction and shape approximation since the resulting representations are typically improper.

Measure computation. A second main achievement from the analysis of the geometric state space is an efficient scheme for measure computation. Without Borchert's framework, the vertices of the polytope must be computed together with a triangulation to obtain, for example, the polytope volume. With the available framework, only a set of linear inequalities must be evaluated followed by the evaluation of a 3^{rd} order polynomial. In this way, the polytope volume computation is quickened by a factor of 10 to 500 which has a considerable impact on the simulation of particle populations. While Borchert focuses on the crystal volume, this present work demonstrates a more general approach towards measure computation.

1.3 Focus and Outline

Thesis. This work focuses on the geometric details for faceted particles and crystals. It aims to connect the problems that appear in the analysis of the geometric state space with concepts and solutions drawn from existing mathematical literature, in particular convex geometry. While important results of Borchert's work have to be derived again to create this connection, a generalization of the existing framework is expected and it is aimed to extend the variety of particle shapes that can be modeled.

Outline. Required mathematical concepts and terminology are introduced in chapter 2 together with outlines of algorithms for geometric problems. Given these preliminaries, new findings and corresponding case studies are organized as follows.

- In chapter 3, the required analysis for the treatment of proper representations is derived based on the introduced mathematical theory. Open aspects of the existing framework are resolved so that any combination of face distances can be handled with the same level of efficiency. A large set of case studies is used to verify the implemented algorithms and to provide data on the related computational effort.
- In chapter 4, a numerical solution scheme for the occurring population balances is derived. The obtained scheme is similar to existing ones but incorporates Monte-Carlo integral estimates so that it is accompanied by a straightforward error estimator and a novel option to optimize the scheme for low numerical errors. The numerical scheme and the error estimator are validated using two case studies for which analytical or semi-analytical results are available.
- In chapter 5, the framework is extended so that abraded particles which exhibit rounded edges can be described. This new shape model is consistent with the

given framework and is rendered possible by the newly introduced mathematical theory. Dynamic shape changes are linked to attrition and growth phenomena, isolating the required parameters to describe the corresponding shape evolution. In addition, a shape identification algorithm based on 2-dimensional projection images is presented and tested in simulation and experiment case studies.

- In chapter 6, two approaches to model reduction and shape approximation are investigated. In the first approach, the dimensionality of the geometric state space is reduced. In the second approach, the shape approximation aims to bypass the complexity of measure computation so that an original population balance model can be reduced to a set of ordinary differential equations. Simulation case studies are used throughout the whole chapter to compare the reduced or approximated results to their original formulations.

The last chapter finally acts as a reminder on the results of this work (the summary) as well as a cheat sheet on the inspirations that could come along while reading (the outlook).

1.4 Literature and Remarks

Overview. The first paragraphs of this subsection illuminate the developments in the field of particle shape modeling, including the listing of relevant literature sources. The last paragraph links to important topics on general crystallization modeling and indicates boundaries and limitations of this work.

Single Crystal Evolution. The main complexity for the geometric evolution of crystal shape comprises the conditions for the appearance and disappearance of crystal faces, given their growth rates. This problem statement was first discussed by Johnsen [1900], who considered conditions for two adjacent faces. It was further developed by considering 2-dimensional crystals, more than two adjacent faces or otherwise constrained approaches [Borgström, 1925, Alexandru, 1969, Szurgot et al., 1991, Prywer, 1992, 1995] until general conditions for a 3-dimensional crystal were provided by Prywer in 1996. All these approaches were static considering only the principle appearance/disappearance conditions. They did not consider the actual dynamic evolution of the crystal. The discussion of such problems was given by Gadewar and Doherty [2004] for prismatic crystals and finally by Zhang et al. [2006] for general 3-dimensional crystals. In both approaches, the evolution of all edges for the current crystal is tracked, given that a face can only disappear when at least 3 edges disappear simultaneously.

Towards Crystal Populations. Approaching particle populations identifies another general issue for the geometric evolution of shape that is the calculation of measures. Based on a review by Cardew [1985], this problem was first mentioned by Christoffersen and Christoffersen [1976] and motivated by the coupling of dissolution kinetics with measurements in batch crystallizers. For such setups, the displacement of specific faces, for which the growth or dissolution laws were studied, cannot be observed directly. They must be retraced by the evolution of the bulk solute concentration. Consequently, several publications consider the relation between the face distances for a crystal population with their volume [Nývlt and Matuchová, 1976a, Christoffersen, 1980, van Oosterhout and van Rosmalen, 1980, Cardew, 1985]. The obtained results, however, are only applicable as long as the particle shape remains similar. This means that the same set

of faces, edges and vertices must remain present at the crystal surface. A much simpler approach is obtained when the shape is assumed to be constant [Nývlt and Matuchová, 1976b]. Such simplified particle populations are simulated by 1-dimensional population balances for which several textbooks are available [Randolph and Larson, 1971, Ramkrishna, 2000].

Shape Evolution of Crystal Populations. Simulating the shape evolution of crystal populations also involves the numerical solution of multidimensional population balances[1]. This additional complexity is discussed in detail in chapter 4 so that the focus on the geometric difficulties is maintained in the following. Ma et al. [2008] studied the growth of potash alum crystals with 3 independent shape variables. It is evident that their results comprise crystals with invalid combinations of face distances[2]. Their later publication [Ma et al., 2008] did also not consider the validity of face distance combinations. In some earlier studies on the solution of 2-dimensional population balances, Ma et al. [2002] or Puel et al. [2003] used geometrically more simple examples for which no unreasonable combinations exist. Additionally, Briesen [2008, 2009a] discussed a comparable approach for the attrition of particles. This time, unreasonable combinations of face distances existed, but they were prevented by the specific choice of the kinetic rates. The first approach that finally considered the full geometric complexity of faceted crystal growth, namely the validity of face distance combinations and efficient measure computation, was given by Borchert [2012].

Borchert's work. Borchert's research on the geometric state space of crystals started with his diploma thesis [Borchert, 2007] which already considers the appearance and disappearance of crystal faces as well as the solution of corresponding population balances. Results of this work were also presented at the *3rd International Conference on Population Balance Modelling* [Borchert et al., 2007] and are incorporated in later scientific papers that solve problem statements of crystallization process design, shape identification from measurements and parameter identification [Borchert et al., 2009, Borchert and Sundmacher, 2011]. Another important step additionally considers the appearance and disappearance of edges at the crystal surface in the analysis of the geometric state space. This allows an efficient approach to measure calculation which was presented at the *17th International Symposium on Industrial Crystallization* [Borchert et al., 2008], but the key components were not published as a full paper until 2012 [Borchert and Sundmacher, 2012]. Note that the first publication from the work presented in this dissertation also falls into that period [Reinhold and Briesen, 2011].

Singh's work. Borchert's diploma thesis was performed at Purdue University at the chair of Prof. Ramkrishna who also acted as the supervisor for Meenesh Singh who submitted his dissertation in 2013. Singh also contributed to the understanding of the geometric state space. He considered crystals with preferred asymmetry while his case study is limited to a 2-dimensional case. However, according to the above history, it is not surprising that the corresponding publication was done together with

[1] This literature review is strictly limited to multidimensional populations balances that consider the growth shape of crystals. Scientific work that considers aggregation or population balances with non-shape-relevant coordinates are beyond the scope of this work. Additional sources may, however, be listed in chapter 4 with respect to particular numerical schemes.

[2] This vague term is only adumbrated in the introduction but clearly defined later in this work or by [Borchert, 2007, 2012].

Borchert [Chakraborty et al., 2010]. In later studies by Singh, he discussed applications of Borchert's framework: shape identification from measurements [Singh et al., 2012] and a utilization and interpretation of the provided geometric framework [Singh and Ramkrishna, 2013, Singh et al., 2013].

Related topics and limitations. This work focuses entirely on the geometric aspects of shape modeling for growth and attrition of 3-dimensional crystals or particles. Nevertheless, related fields of interest are listed in the following to indicate these limitations more specifically and provide relevant literature for further studies. Most of the aspects below are generally covered by industrial crystallization handbooks like Mullin [2001] or Mersmann [1994] or also the work of Briesen [2008].

The molecular lattice of a crystal and its symmetry constitutes the structural origin for the polyhedral shapes that are assumed by crystals [Borchardt-Ott, 2009, Sands, 1993]. This does also include the common notation of crystal faces by Miller indices like the ones that are used for the figure 1.1 on page 2 (*e.g.* $\{110\}$).

The dynamic aspects for crystal shape are given by the rates of nucleation [Lacmann et al., 1999, Erdemir et al., 2009] and growth [Lacmann et al., 1999, Boerrigter, 2003]. Nucleation is mentioned here because the lattice structure of the nucleus cannot be changed by growth which is important for enantiomers. In regards to growth, Taylor et al. [1992] provide an interesting overview of a mathematically more general approach to the motion of interfaces. This approach merely assumes a dependence of the velocity on the volumetric and surface based free energy and it turns out that the typically assumed polyhedral growth model is only one out of nine discussed approaches. The counterpart of growth is dissolution [Snyder et al., 2007, Snyder and Doherty, 2007, Singh, 2013]. When this process is not diffusion limited, it is also covered by the geometric growth models given by Taylor et al. [1992]. Otherwise, the resulting equations are called Stefan problems [Chen et al., 1997, Vermolen et al., 2005] and, to the author's knowledge, the resulting crystal shapes are not yet considered for the modeling of particle populations.

Attrition is covered in detail in chapter 5 while breakage [Hill, 2004] or aggregation [Bramley et al., 1996] of particles is not treated. While breakage creates reasonable convex particles, the generated particle shapes do not correlate well to the shape model for faceted growth. Aggregated particles are clearly not convex and therefore beyond the scope of this framework which is developed from convex geometry.

2

Geometry

2.1 Introduction and Preliminaries

2.1.1 Introduction

Convex Geometry. In this work, the theory of convex bodies is incorporated into the analysis of the geometrical state space of faceted particles. The fundamentals of convex geometry as a separate field of mathematics or geometry are given by two independent developments of Karl Hermann Brunn [1887] and Hermann Minkowski [1911] [Kjeldsen, 2008]. Therefore, this field is also denoted by the term 'Brunn-Minkowski theory'. Since its first applications to number theory at the end of the 19$^{\text{th}}$ century, it has nowadays influenced numerous other fields of interest like optimization, graph theory or coding theory [Gruber, 2004]. General textbooks on convex geometry are given by Bonnesen and Fenchel [1934], Hadwiger [1957], Grünbaum [2003] and, in particular, Schneider [2008]. This field provides the core of the mathematical theory while supplemental material is drawn from integral geometry [Santaló, 1976] and combinatorial geometry [Edelsbrunner, 1987]. Especially important is also the field of computational geometry [Preparata and Shamos, 1985, Gritzmann and Klee, 1993, O'Rourke, 1994] from which the applied algorithms originate.

Wulff theorem. While convex geometry is hardly known in crystallization engineering, it was already applied for the more commonly known Wulff theorem [Wulff, 1901, Fonseca, 1991]. The Wulff theorem states that the surface free energy of a fixed amount of crystalline substance is minimized by exactly that shape for which the face distances of all possible faces are proportional to the corresponding surface free energy of that direction. Note that the surface free energy is dependent on the direction of a crystal face since the underlying molecular lattice generates different surface configurations. In other words, the Wulff theorem is required to predict the thermodynamical equilibrium shape of crystals. This theorem was proven by the application of the so-called Brunn-Minkowski inequality which is an analogon to the triangle inequality.[1] The development of this proof with successively increasing generality is described in [Herring,

[1] Describing the Brunn-Minkowski inequality is beyond the scope of this work. But assuming the knowledge that is provided in this chapter, it can be explained as follows. For two convex bodies $A, B \in \mathbb{R}^n$ and the Minkowski addition $\lambda A + (1-\lambda)B$, the following inequality for the volume measure μ^{volume} holds: $\mu^{\text{volume}}(\lambda A + (1-\lambda)B) \geq (1-\lambda)\sqrt[n]{(\mu^{\text{volume}}(A))} + \lambda\sqrt[n]{(\mu^{\text{volume}}(B))}$.

1951], [Fonseca, 1991] or Singh [2013]. The theorem was also adopted by Chernov [1963] to predict the steady-state morphology of a crystal based on predicted face specific growth rates [Gadewar and Doherty, 2004].

Overview. The remainder of this subsection contains some general remarks on notation while subsection 2.1.2 introduces basic terms for geometric objects. Section 2.2 then presents two shape representations that are commonly used in mathematical literature and one shape representation that is newly defined for the scope of this work. All corresponding definitions are prerequisites for the derivations and discussions in subsequent parts of this work. Section 2.3 explains principal concepts for measure computation and the two algorithms in section 2.4 are central for the derivations of chapter 3.

Physical objects and state spaces. The notation and definitions introduced in this and the following sections are used for geometric objects that are 1-, 2- or 3-dimensional as well as for sets of vectors as they occur for state spaces. The intuitive difference between both is that measures like length, distance and angles are commonly applied for physical objects while such measures usually are not even required for state spaces. However, the mathematical theory of both objects that is linear algebra is almost the same [Lorenz, 2003, 2005, Bronstein et al., 2008]. Real vector spaces $\mathbb{R}^n$ that are commonly used for state spaces can be extended to the Euclidean vector space $\mathbb{E}^n$ that is commonly used for geometric objects and for which measures like length, distance and angle are defined.

Euclidean vector space. The n-dimensional real vector space $\mathbb{R}^n$ only implies vector addition and scalar multiplication [Lorenz, 2003]. To obtain measures like the diameter or the volume of geometric objects, the Euclidean vector space, or simply Euclidean space, $\mathbb{E}^n$ is required which implies that the terms length, distance and angle are defined.

For the vectors $\mathbf{u} = (u_1, \dots, u_n)^T$ and $\mathbf{v} = (v_1, \dots, v_n)^T$ in Euclidean space ($\mathbf{u}, \mathbf{v} \in \mathbb{E}^n$) the scalar product is defined by:

$$\langle \mathbf{u}, \mathbf{v} \rangle = \mathbf{u} \bullet \mathbf{v} = \sum_i u_i v_i. \tag{2.1}$$

The notation $\langle \mathbf{u}, \mathbf{v} \rangle$ is common in the context of geometry while the notation $\mathbf{u} \bullet \mathbf{v}$ is used alternatively in the context of analysis (*e.g.* $\operatorname{div} \mathbf{f}(\mathbf{x}) = \nabla_{\mathbf{x}} \bullet \mathbf{f}(\mathbf{x})$). The length of the vector $\mathbf{v}$ is defined by the Euclidean norm:

$$\|\mathbf{u}\| = \sqrt{\langle \mathbf{u}, \mathbf{u} \rangle}. \tag{2.2}$$

The distance between two vectors or points is given by $\|\mathbf{u} - \mathbf{v}\|$ and the non-reflective angle $0° < \theta < 180°$ between the two vectors is given by:

$$\theta = \arccos \frac{\langle \mathbf{u}, \mathbf{v} \rangle}{\|\mathbf{u}\| \, \|\mathbf{v}\|}. \tag{2.3}$$

It is assumed that any real vector space can be extended to an Euclidean vector space so that $\mathbb{E}^n$ and $\mathbb{R}^n$ are considered equivalent in subsequent sections of this work. This also implies that for most of the upcoming definitions the term 'geometric object' can be replaced by 'set of vectors' and *vice versa*.

Sets of points and sets of vectors. Geometric objects are represented by sets of points or vectors. While a vector describes the directed path between two points, it will not be distinguished between points and vectors. A point is equivalent to the corresponding vector pointing from the origin to that point. In the context of geometry and vector spaces, bold lower-case letters are applied for points or vectors (*e.g.* $\mathbf{x}$, $\mathbf{v}$ or $\mathbf{u}$) and upper-case letters for sets of points or vectors (*e.g.* A or B). Treating geometric objects as sets of vectors immediately allows to use the notation that is common for sets and vector spaces. Typical set operations might be set union ($\cup$), set intersection ($\cap$) or relative complement ($\setminus$).

2.1.2 Geometric Objects

Geometric bodies. Geometric objects might, in general, be any subset of the Euclidean vector space. However, all physical bodies considered in this work will be bounded and closed subsets[2] of the Euclidean space. This excludes, for example, fractals which are not closed. An object of this class (bounded and closed) is called a geometric body or n-dimensional body.

Convexity. The geometric object B is called convex if for any two elements $\mathbf{x}$ and $\mathbf{y}$ of B, all elements of the closed line segment $[\mathbf{x}, \mathbf{y}]$ are part of the geometric object B, or formally written:

$$B \text{ is convex } \Leftrightarrow (\mathbf{x}, \mathbf{y} \in B, \mathbf{z} \in [\mathbf{x}, \mathbf{y}] \Rightarrow \mathbf{z} \in B)\,. \tag{2.4}$$

This defines convexity for any subset of a vector space, not just bodies. Such a convex set is not necessarily bounded nor closed.

Unit sphere, unit ball and unit vectors. Vectors for which only the direction is relevant are usually normalized to have the length one. The set of all unit vectors in $\mathbb{R}^n$ is called the unit $(n-1)$-sphere and is denoted by S^{n-1}. If an n-dimensional vector $\mathbf{v}$ of an arbitrary direction is required, the notation $\mathbf{v} \in S^{n-1}$ is used. The unit $(n-1)$-sphere is the surface of the unit n-ball and, hence, is itself $(n-1)$-dimensional while the points exist in the n-dimensional space. The unit n-ball is denoted by B^n.

Examples are given to clarify the terms above. The line segment $[-1,\, 1]$ is the unit 1-ball and the corresponding unit 0-sphere consists of the points -1 and 1. The unit 2-ball is a disk with radius 1 and center in the origin. Its circumference is the corresponding 1-sphere.

Convex hull, positive hull & cones. The terms 'convex hull' and 'positive hull' are common operators in mathematical literature. Figure 2.1 shows the convex hull and the positive hull of a point set B. The point set B consists of the four black points around the label $\mathrm{conv}(B)$. The convex hull (the shape labeled with $\mathrm{conv}(B)$) is a convex body and, hence, bounded while the positive hull (the cone labeled with $\mathrm{pos}(B)$, including the convex body) is unbounded.

In general, given a nonempty set B, the convex hull of B is defined by:

$$\mathrm{conv}\, B = \{\sum_i \lambda_i \mathbf{b}_i \mid \mathbf{b}_i \in B,\, \lambda_i \in \mathbb{R},\, \lambda_i \geq 0,\, \sum_i \lambda_i = 1\} \tag{2.5}$$

[2] Bounded sets do not extent to infinity. For closed sets, the boundary belongs to the set. For example, all numbers $x \in \mathbb{R}^1$ with $0 \leq x \leq 1$ are an bounded and closed set. The numbers with $0 < x \leq 1$ determine a bounded and open set.

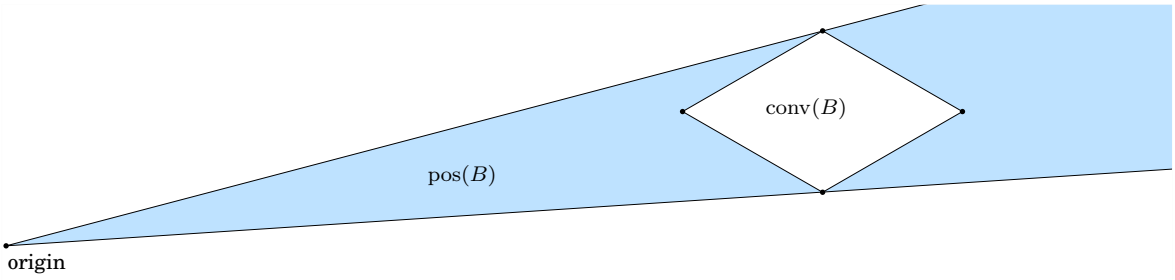

Figure 2.1.: A set B (black dots except for the origin), the convex hull of B (white shape) and the positive hull of B (light blue shape including the white shape).

and the positive hull by:

$$\operatorname{pos} B = \{\sum_i \lambda_i \mathbf{b}_i \mid \mathbf{b}_i \in B,\ \lambda_i \in \mathbb{R},\ \lambda_i \geq 0\} \tag{2.6}$$

while the total number of elements $\mathbf{b}_i$ that is combined for $\operatorname{conv} B$ and $\operatorname{pos} B$ is arbitrary but finite [Schneider, 2008, page 1 and 2]. A convex cone, or simply cone, is further defined by any nonempty set C for which each element $\mathbf{c} \in C$ implies that $\lambda\mathbf{c} \in C$ holds for any $\lambda \geq 0$, $\lambda \in \mathbb{R}$. The positive hull always creates an convex cone while convex cones are frequently used for regions of the geometric state space (see chapter 3).

Hyperplane and halfspace. A hyperplane is the generalization of a plane in 3-dimensional space. It can be defined by a normal vector $\mathbf{a} \in \mathbb{R}^n$ and a scalar $h \in \mathbb{R}$:

$$H(\mathbf{a},\, h) = \{\mathbf{x} \mid \langle \mathbf{a},\, \mathbf{x} \rangle = h\}, \tag{2.7}$$

where h is the distance of the plane from the origin if the vector $\mathbf{a}$ is a unit vector. A hyperplane is always $(n-1)$-dimensional and separates the corresponding vector space into two parts. Both parts of the vector space, including the hyperplane, represent a halfspace. Hence, a halfspace is a subset of the $\mathbb{R}^n$ that is defined by:

$$H^-(\mathbf{a},\, h) = \{\mathbf{x} \mid \langle \mathbf{a},\, \mathbf{x} \rangle \leq h\}. \tag{2.8}$$

This definition implies that the normal vector $\mathbf{a}$ points to the outside of the halfspace.

Support function. The support function is an important tool for the Brunn-Minkowski theory and will therefore be used as the basis for several definitions in this section. The support function assigns each convex body B a uniquely determined function $h(B, \cdot)$: $\mathbb{R}^n \mapsto \mathbb{R}$.[3] The support function is defined for general geometric objects B by [Schneider, 2008, section 1.7]:

$$h(B,\, \mathbf{v}) = \sup_{\mathbf{b} \in B} \langle \mathbf{b},\, \mathbf{v} \rangle\,, \tag{2.9}$$

where the supremum (sup), the least upper bound of the argument, is identical to the maximum (max) for geometric bodies[4]. As vectors can be interpreted as points, the

[3] This function might be interesting for shape approximation because knowledge of approximating functions could be utilized to approximate convex bodies.

[4] The supremum is only required in the case of non-compact sets. For example, the support function of a fractal would require the application of the supremum.

support function assigns every point $\mathbf{v}$ in space a scalar value $h(B, \mathbf{v})$ according to the convex body. A 2-dimensional example is shown in figure 2.2 (left). The polygon is drawn in blue. The support function that has values for any point in the visible coordinate range is presented by the labeled contour lines for $h(B, \mathbf{v}) = 1, 2, 3, 4$ and 5.

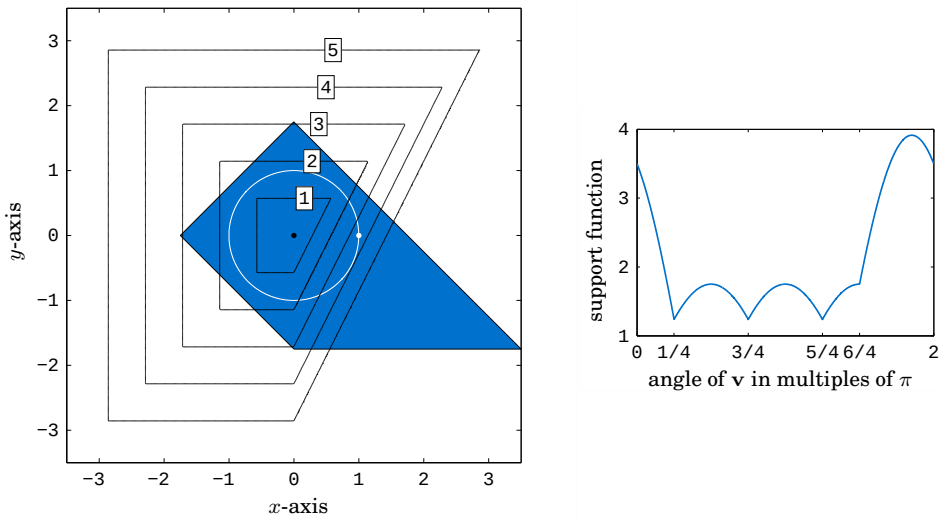

Figure 2.2.: Support function of a polygon; left: the polygon in blue and the contour lines of the support function in black with labels of the corresponding support function values; right: polar plot of the support function $h(B, \mathbf{v})$ for $\mathbf{v} \in S^1$ where the x-axis is the angle between the vector $\mathbf{v}$ and the x-axis (path drawn in left figure in white)

Support function as a representation. It is important to note that for $\lambda \geq 0$, $\lambda \in \mathbb{R}$ it holds:

$$h(B, \lambda\mathbf{v}) = \lambda\, h(B, \mathbf{v}) = h(\lambda B, \mathbf{v}) \tag{2.10}$$

so that using the unit vectors $\mathbf{u} \in S^{n-1}$ is sufficient to characterize the support function. According to the definition, equation 2.9, the support function $h(B, \mathbf{u})$ represents the signed distance of the point $\mathbf{b} \in B$ that is farthest away from the origin in the direction of $\mathbf{u}$ so that $h(B, \mathbf{u})$ defines the outer hull of the convex body B. In consequence, knowing the support function $h(B, \mathbf{u})$ with $\mathbf{u} \in S^{n-1}$ is sufficient to define a convex body B [Schneider, 2008, theorem 1.7.1]. In 2-dimensions, any convex body can be represented by a function $\mathbb{R} \mapsto \mathbb{R}$ that maps, for example, the angle between the unit vector and the x-axis to the support function $h(\cdot, \mathbf{u})$. Figure 2.2 (right) shows an example for such a function which represents the polygon in the left part of the figure. For a 3-dimensional convex body, a function $\mathbb{R}^2 \mapsto \mathbb{R}$ is required whose arguments could be the polar angles of the unit vector.

Supporting hyperplanes, halfspaces and sets. Figure 2.3 gives two examples for the entities that are introduced in the following. These entities are based on a polytope which is colored in blue and a specific direction $\mathbf{u}$. In the left example, the direction coincides with a facet normal of the polytope. The light line indicated by $H(B, \cdot)$ is the

support hyperplane as the hyperplane $H(\mathbf{u},\, h)$ for which h is minimal while the polytope is still contained in the halfspace $H(\mathbf{u},\, h)$ (region drawn in light blue). This halfspace is called the supporting halfspace $H^-(B, \cdot)$. The intersection of the hyperplane and the polytope is called the support set $F(B, \cdot)$ which is identical to an edge of the polytope for the left example. In the right example, a different direction $\mathbf{u}$ is chosen and the support set $F(B, \mathbf{u})$ is identical to an extreme point of the polytope.

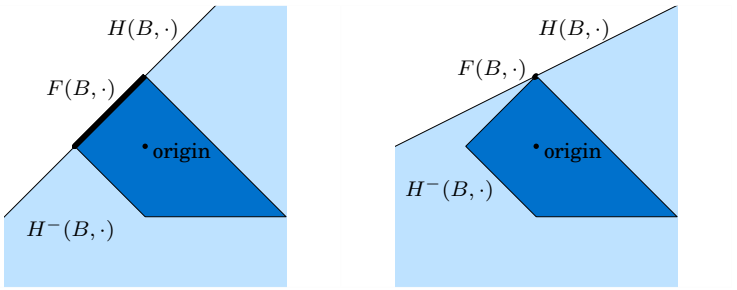

Figure 2.3.: Support hyperplane (light line, $H(B, \cdot)$), halfspace (light blue region, $H^-(B, \cdot)$) and set (bold line or dot, $F(B, \cdot)$ for a polygon (blue) and two different directions (left/right)

In general, the support hyperplanes $H(B,\, \cdot)$, supporting halfspaces $H^-(B,\, \cdot)$ and the support sets $F(B,\, \cdot)$ are defined for a geometric body $B \subset \mathbb{R}^n$ and a vector $\mathbf{v} \in \mathbb{R}^n$ by the application of the support function:

$$H(B,\, \mathbf{v}) = \{\mathbf{x} \mid \langle \mathbf{x},\, \mathbf{v} \rangle = h(B,\, \mathbf{v}),\, \mathbf{x} \in \mathbb{R}^n\} \tag{2.11}$$

$$H^-(B,\, \mathbf{v}) = \{\mathbf{x} \mid \langle \mathbf{x},\, \mathbf{v} \rangle \leq h(B,\, \mathbf{v}),\, \mathbf{x} \in \mathbb{R}^n\} \tag{2.12}$$

$$F(B,\, \mathbf{v}) = H(B,\, \mathbf{v}) \cap B. \tag{2.13}$$

Note that the length of the vector $\mathbf{v}$ is irrelevant so that for any $\lambda > 0,\, \lambda \in \mathbb{R}$ it holds: $H(B,\, \lambda\mathbf{v}) = H(B,\, \mathbf{v})$, $H^-(B,\, \lambda\mathbf{v}) = H^-(B,\, \mathbf{v})$ and $F(B,\, \lambda\mathbf{v}) = F(B,\, \mathbf{v})$. Any supporting hyperplane exactly touches the corresponding convex body. Any convex body B equals the intersection of all supporting halfspaces: $B = \bigcap_{\mathbf{u} \in S^{n-1}} H^-(B,\, \mathbf{u})$. And the union of all support sets exactly represents the boundary of the corresponding polytope: $\partial B = \bigcup_{\mathbf{u} \in S^{n-1}} F(B,\, \mathbf{u})$.

Face, facet and extreme point. The term 'face' for a 3-dimensional polytope is commonly understood as one of the 2-dimensional support sets. In the context of convex geometry, the term 'face' has a more general meaning while 'facet' and 'extreme point' are two more terms to be considered. 'Face' and 'facet' are used differently dependent on the chosen literature and the terminology here is adopted from [Schneider, 2008, section 2.1]. Any d-dimensional support set $F(B, \cdot)$ of the convex body $B \subset \mathbb{R}^n$ is called a d-face of B. For a 3-dimensional polytope, like the crystals to be considered, any edge is a 1-face, any vertex is a 0-face and any 2-dimensional support set is a 2-face. A facet denotes a $(n-1)$-face and an extreme point is any 0-face of a convex body. Extreme points of a polytope will be denoted by: $\mathbf{x}_{\mathrm{E},i}$.

2.2 Shape Representation

2.2.1 $\mathcal{H}$-Representation

Motivation. A mathematical description of a convex body must exist so that they can be modeled and analyzed algorithmically. This and the following subsection introduce two representations for convex bodies that are frequently used in computational geometry [Gritzmann and Klee, 1993, Hufnagel, 1995]. Both representations describe the same subset of convex bodies that are commonly understood as polytopes or polyhedra. The representation in this section is of major importance for the concepts in the upcoming chapters so that an extensive notation is introduced here.

Definition. The intersection of a finite number of halfspaces is called a polyhedron. If this polyhedron is bounded it represents a convex body. The corresponding set of halfspaces that are defined by normal vectors $\mathbf{a}_i$ and scalars h_i is the $\mathcal{H}$-representation of the polyhedron [Hufnagel, 1995, page vii]. In the scope of this work, the normal vectors $\mathbf{a}_i$ are usually fixed and different polyhedra are obtained by varying the scalars h_i. The total number of normal vectors is denoted by n_H, the normal vectors are collected in a matrix:

$$\mathbf{A} = (\mathbf{a}_1,\, \mathbf{a}_2,\, \ldots,\, \mathbf{a}_{n_\mathrm{H}})^T = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n_\mathrm{H},1} & a_{n_\mathrm{H},2} & \cdots & a_{n_\mathrm{H},n} \end{bmatrix}, \tag{2.14}$$

and the scalars h_i are collected in a vector $\mathbf{h} = (h_1,\, h_2,\, \ldots,\, h_{n_\mathrm{H}})^T$ so that a corresponding polyhedron is denoted by $P_\mathbf{A}(\mathbf{h})$ or simply $P(\mathbf{h})$ if the context clearly defines the matrix $\mathbf{A}$. The definition above can be formulated directly as (see equation 2.8):

$$P_\mathbf{A}(\mathbf{h}) = \bigcap_i H^-(\mathbf{a}_i,\, h_i). \tag{2.15}$$

However, usually the following equivalent expression is applied:

$$P_\mathbf{A}(\mathbf{h}) = \{\mathbf{x} \mid \mathbf{A}\,\mathbf{x} \leq \mathbf{h},\, \mathbf{x} \in \mathbb{R}^n\}\,. \tag{2.16}$$

Throughout this work the vectors $\mathbf{a}_i$ are unit vectors so that the scalars h_i can be called facet distances and the vector $\mathbf{h}$ is called the vector of facet distances. In general, the facet distances h_i might be negative so that for these cases h_i is not equivalent to the distance as defined with the Euclidean norm.

Note that the facet distance h_i equals the value of the support function for the direction of the corresponding normal vector: $h(P(\mathbf{h}),\, \mathbf{a}_i) = h_i$ so that the nomenclature is consistent.[5] In consequence, the halfspaces (or hyperplanes) that define a polyhedron are typically identical to the supporting halfspaces (or hyperplanes).

Example. An example of a polyhedron in 2-dimensional space is drawn in figure 2.4. This example is also used to demonstrate how a possible physical dimension of a geometric object can be reflected in a $\mathcal{H}$-representation. The facet normals are outer normals

[5] Considering the optimization problem that defines the support function (equation 2.9), the active linear constraint is: $\mathbf{a}_i\mathbf{x} \leq h_i$ (see equation 2.16). Hence, the support value is $\langle \mathbf{a}_i, \mathbf{x} \rangle = h_i$.

because of the definition of a halfspace. The matrix of facet normals is given by:

$$\mathbf{A} = \begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ -1/\sqrt{2} & 1/\sqrt{2} \\ -1/\sqrt{2} & -1/\sqrt{2} \\ 0 & -1 \end{pmatrix}. \tag{2.17}$$

The values h_i represent the distance of the corresponding facet from the origin (drawn for h_4) and might therefore carry the physical dimension of the polyhedron. The vector of facet distances is given by:

$$\mathbf{h} = 1.3 \times \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 1 \right)^T \text{cm}. \tag{2.18}$$

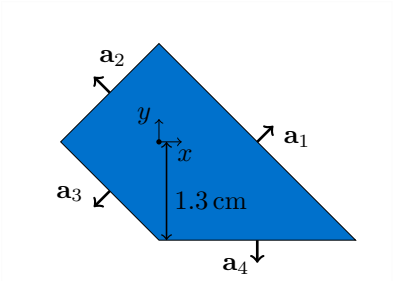

Figure 2.4.: A polyhedron in blue in $\mathcal{H}$-representation (equations 2.17 and 2.18); normal vectors drawn with bold arrows; the double arrow marks the facet distance h_4

2.2.2 $\mathcal{V}$-Representation

Definition. Any finite set of points B is a $\mathcal{V}$-representation of the corresponding convex hull ($\operatorname{conv} B$). However, the term $\mathcal{V}$-representation is usually associated with a minimal set of points that describe the given polytope. These points are the extreme points of the polytope and the number of extreme points that are used for a $\mathcal{V}$-representation is denoted by n_E.

Polytopes and polyhedra. While a $\mathcal{H}$-representation defines polyhedra that are not necessarily bounded, a $\mathcal{V}$-representation defines polytopes that are bounded by definition. However, if a polyhedron is bounded, the term polytope can be used synonymously so that the usage of the term polyhedron or polytope is usually chosen according to the associated representation.

Simplex and simplicial cone. The minimal number of vertices that describe a d-dimensional polytope is given by $d + 1$. Such a polytope is generally denoted as a d-simplex and might be embedded in some n-dimensional space ($n \geq d$). The n-simplex is simply called a simplex. For example, any triangle in 2-dimensions is a simplex. The term simplicial cone refers likewise to an n-dimensional cone in n-dimensional space that is generated by a positive hull from exactly n points. Note that instead of d extreme points for a $\mathcal{V}$-representation, d facets of a $\mathcal{H}$-representation could be used.

2.2.3 Basic Operations and Minkowski Addition

Motivation. Given normal vectors and facet distances ($\mathcal{H}$-representation) or extreme points ($\mathcal{V}$-representation), these scalars and vectors are used to describe the changes in shape and size for convex bodies. This restricts the set of bodies that can be handled to polyhedra or polytopes, respectively. This section defines operations that change shape, size or position for general geometric objects. Based on these operations, an alternative shape representation will be introduced in subsection 2.2.4 that is much less common and has never been applied to crystals.

Basic operations. Given a geometric object B, the addition with a vector $\mathbf{v} \in \mathbb{R}^n$ is defined by:

$$B + \mathbf{v} = \{\mathbf{x} + \mathbf{v} \mid \mathbf{x} \in B\}. \tag{2.19}$$

This addition can be considered as displacing the object B by the vector $\mathbf{v}$ so that only the position of that object is changed.

Multiplication by a scalar $\lambda \in \mathbb{R}$ is defined by the multiplication of its elements:

$$\lambda B = \{\lambda\mathbf{x} \mid \mathbf{x} \in B\}. \tag{2.20}$$

This scalar multiplication can be considered as changing the size of the object B, even though the position of the object might also change if it is not the origin.

Minkowski addition. The Minkowski addition is defined for arbitrary sets $A, B \subset \mathbb{R}^n$ with [Schneider, 2008, page 126]:

$$A + B \quad = \quad \{\mathbf{a} + \mathbf{b} \mid \mathbf{a} \in A,\, \mathbf{b} \in B\}\,. \tag{2.21}$$

The convex sets A and B are called summands of the resulting set $(A + B)$.

Figure 2.5 demonstrates the Minkowski addition of an octahedron and a cube. Positions of the objects are ignored, because only size and shape are the significant properties for the framework that will be introduced in this work. The Minkowski sum is obtained by taking an arbitrary point of the cube (marked with a black dot) and move it with the cube along all points of the octahedron. The space covered by this procedure is the Minkowski sum of both polytopes. This perception is represented by an alternative definition of the Minkowski addition:

$$A + B = \bigcup_{\mathbf{b} \in B} A + \mathbf{b} \tag{2.22}$$

that is equivalent to equation 2.21.

The Minkowski addition is sometimes referred to as vector addition in mathematical literature. In fact, the following equalities hold ($A, B \subset \mathbb{R}^n$, $\lambda, \mu \in \mathbb{R}$):

$$A + B = B + A \tag{2.23}$$
$$A + (B + C) = (A + B) + C \tag{2.24}$$
$$\lambda(A + B) = \lambda A + \lambda B \tag{2.25}$$
$$(\lambda + \mu)A = \lambda A + \mu A \tag{2.26}$$

so that the common understanding of the calculus in vector spaces can be applied, even though the Minkowski addition together with the scalar multiplication does not form a

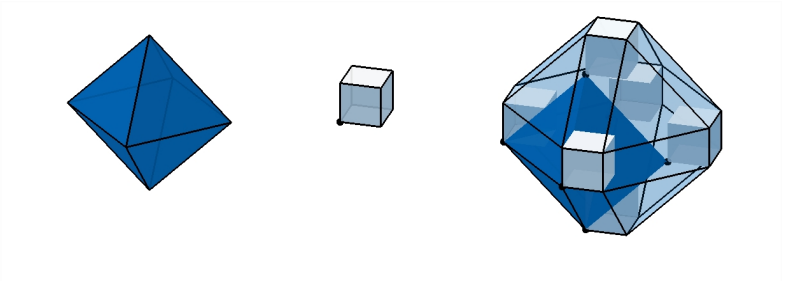

Figure 2.5.: Octahedron (blue, left), cube (light blue, middle) and Minkowski addition of both polytopes (right).

proper vector space. The neutral element for the addition is $\{\mathbf{0}\}$, but there is no inverse element '$-A$' so that $A + (-A) = \{\mathbf{0}\}$. However, as long as no negative scalar coefficients are used the common intuition from vector spaces will be valid. In particular, the Minkowski difference[6] $A - B$ is a separate operation that is defined by:

$$A - B = \bigcap_{\mathbf{b} \in B} A - \mathbf{b}.$$

Furthermore, the scalar multiplication $-1 \cdot A$ represents the point reflection of A. This implies that $A + (-1 \cdot A)$ would be perceived as a larger body than A and $A + (-1 \cdot A)$ is not equal to the Minkowski difference $A - A$ which would result in $\{\mathbf{0}\}$.

Minkowski addition and support function. The Minkowski addition of two convex bodies A and B has an interesting and important relation to the support function of these polytopes (see equation 2.9). This relation is [Schneider, 2008, theorem 1.7.5]:

$$h(A + B, \cdot) = h(A, \cdot) + h(B, \cdot) \tag{2.27}$$

and reads as: the support function of the Minkowski sum is the sum of the support functions of the corresponding summands. Strongly connected to this and very useful to understand Minkowski addition is that for two convex bodies any support set of the sum equals the Minkowski addition of the support sets of both separate polytopes (in the same direction):

$$F(A + B, \cdot) = F(A, \cdot) + F(B, \cdot). \tag{2.28}$$

This implies for 3-dimensional polytopes in particular that any face of a summand must be a face of the sum and any edge of a summand must be an edge or face of the sum. This can be formally written as: $\dim F(A + B, \cdot) \geq \max(\dim F(A, \cdot), \dim F(B, \cdot))$. Furthermore, if the support set of the sum is an edge, its length is determined by the sum of the lengths of the corresponding edges of the summands.

2.2.4 $\mathcal{S}$-representation

Definition. Given the operations for geometric bodies from the preceding subsection, it is now possible to define a new representations that uses a fixed set of arbitrary bodies as the generating elements instead of points ($\mathcal{V}$-representation) or hyperplanes

[6] Minkowski difference is also denoted as Pontryagin difference in the context of control theory [Kvasnica et al., 2004].

($\mathcal{H}$-representation). The (ordered) set of bodies is denoted by $\mathcal{S} = \{S_1, \ldots, S_{n_S}\}$ while n_S denotes the total number of bodies S_i. The represented shapes are then generated as linear combinations with arbitrary scalar coefficients $\lambda_i \in \mathbb{R}$, $\lambda_i \geq 0$ that can be arranged in a vector $\boldsymbol{\lambda}$:

$$P_{\mathcal{S}}(\boldsymbol{\lambda}) = \sum_i \lambda_i S_i. \tag{2.29}$$

Such representations will be called $\mathcal{S}$-representations and the bodies S_i are referred to as structuring elements. The subscript $\mathcal{S}$ for $P_{\mathcal{S}}$ is typically omitted in applications. In contrast to $\mathcal{V}$- or $\mathcal{H}$-representations, the bodies S_i and, hence, also the represented shapes $P_{\mathcal{S}}(\boldsymbol{\lambda})$ are not restricted to polytopes or polyhedra. In chapter 5, this feature will be utilized to model rounded particles.

Literature. The above concept (but not the notation) is already denoted by Brunn [1887] as 'convex linear family of convex bodies' (original: *'Konvexe Linearschar konvexer Körper'*) for $n_S = 2$ [Blaschke, 1916] and is the basis of the framework presented here. The Minkowski addition is also used in morphological image analysis in which one summand is the image and the other summand is a filter that is called structuring element [Szoplik, 1996]. This name is chosen for this work because it reflects the intrinsic nature of the Minkowski addition: the morphological structure of the sum is determined by the morphological structure of the structuring elements with coefficients $\lambda_i \neq 0$. The benefits of this representation become evident in subsection 2.3.2.

2.3 Polytope Measures

2.3.1 Volume Computation

Motivation. An important part of the complexity that is connected with the modeling of crystal shape comes with the necessity of volume calculation. Determining the total volume of a crystal population is mandatory for the simulation of saturation dependent crystal growth. Additionally, calculating any other measure relies on volume computation in dimension three or less. Beyond that, the volume is used as the primal example for measures in general.

Spectrum of possible methods. An excellent overview of deterministic methods for volume computation is given by Gritzmann and Klee [1993, section 3]. Other textbooks on computational geometry or algorithmical geometry, *e.g.* O'Rourke [1994], are, however, equally helpful. The most important method is *triangulation* which will be explained in the following paragraph. All other methods do not have any benefit over this approach [Gritzmann and Klee, 1993]. Nevertheless, two alternatives are named to outline the range of methods. The 'sweeping-plane formula' does not explicitly involve triangulation. However, if it is applied for general polytopes, a triangulation is implicitly contained. 'Numerical integration' is well known and includes Monte-Carlo and quasi-Monte-Carlo methods [Press et al., 2007, Kroese et al., 2011]. These methods are not very efficient for volume calculation, even though volume calculation poses the historical origin of numerical integration [Gritzmann and Klee, 1993, section 3.4].

Triangulation. Figure 2.6 shows the triangulations of two polytopes that share the same set of facet normals. Only the triangulation of two representative facets is shown. Triangulation is defined as the dissection of a polytope into simplices for which the intersection of any two simplices is a face of each.[7] Once a polytope is triangulated into simplices, the volume calculation is straightforward and is comprised of the sum of the simplex volumes. The volume of any n-dimensional simplex is calculated by:

$$\mu_{\text{simplex}}^{\text{volume}} = \frac{1}{n!} \left| \det \left(\mathbf{x}_{\text{E},2} - \mathbf{x}_{\text{E},1},\ \mathbf{x}_{\text{E},3} - \mathbf{x}_{\text{E},1},\ \ldots, \mathbf{x}_{\text{E},n+1} - \mathbf{x}_{\text{E},1} \right) \right| , \tag{2.30}$$

where $\mathbf{x}_{\text{E},i}$ are the $(n+1)$ extreme points of the simplex.

Applied algorithm (Qhull). For the implementation of volume calculation, the quickhull algorithm [Barber et al., 1996] is used, which is readily available in MATLAB® [2012]. This algorithm calculates the triangulated convex hull for a set of points. However, instead of the polytope, the surface of the polytope is triangulated. The triangulation of the polytope is obtained by choosing any interior point of the polytope. The triangulations in figure 2.6 are obtained that way. The algorithm simultaneously generates the facets of the convex hull so that the output is a $\mathcal{V}$-representation, as well, as a $\mathcal{H}$-representation. However, the $\mathcal{H}$-representation cannot be accessed in MATLAB®[8].

[7] A face might be empty for empty intersections. This is the definition used in literature. As an alternative, triangulation could be defined by the dissection of a polytope into simplices for which the volume of the intersection of two different simplices must be 0. This definition would, however, not be equivalent since it allows dissections which are not possible with the original definition. The condition that each pair of two simplices must be disjoint might also be more intuitive but is not applicable since the simplices can share the same facets or edges.

[8] MATLAB® calls the original implementation but does not copy the results of the $\mathcal{H}$-representation into its workspace.

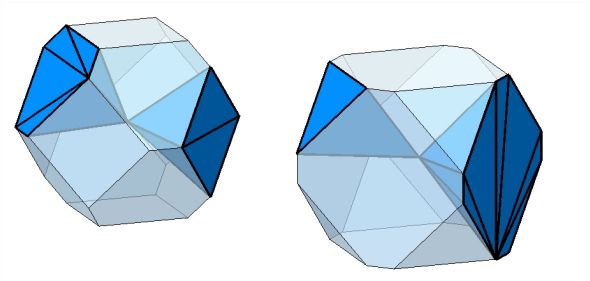

Figure 2.6.: Triangulation of two polytopes with the same set of facet normals (facets of a cube and facets of an octahedron)

Barber et al. [1996] also give an empirical comparison to other algorithms and they discuss practical considerations for degenerated input and imprecision of round-off errors.

Complexity for crystal modeling. A $\mathcal{H}$-representation with a fixed set of facet normals but with a varying vector of facet distances is assumed which is the typical setup for crystallization modeling. As long as the same edges remain present and only change in length, each extreme point $\mathbf{x}_{\mathrm{E},i}$ from above is defined by the intersection of the same 3 facets so that the extreme points are a linear function of the vector of facet distances. Furthermore, a valid triangulation is achieved by combining the same extreme points to simplices even when the overall shape changes. Following equation 2.30, the volume is a 3^{rd} order polynomial of the extreme points. The result is an equation that is commonly known in mathematical literature: given the vector of facet distances $\mathbf{h}$, the volume of a polytope can be calculated, by [Schneider, 2008, lemma 5.1.2]:

$$\mu^{\mathrm{volume}}(\mathbf{h}) = \sum_{i,j,k} c^{\mathrm{volume}}_{p,(i,j,k)} h_i h_j h_k, \tag{2.31}$$

where the set of coefficients $c^{\mathrm{volume}}_{p,(i,j,k)}$ is valid as long as the polytope comprises the same set of edges. Hence, p is an index to a specific set of edges being present at the polytope surface.

a-types. Based on the discussion above, the combinatorial structure of the polytope surface is an important property for measure computations. Polytopes that are similar in such a manner were called analogue in early mathematic literature so that the class of polytopes with the same combinatorial surface structure are called an a-type [Meyer, 1974, Schneider, 2008]. This term will be defined in subsection 2.4.1 and is frequently used throughout this work. Note that the total number of a-types is finite since the set of facet normals is fixed for crystals so that only a finite number of coefficient sets $c^{\mathrm{volume}}_{p,(i,j,k)}$ exists for equation 2.31. The partitioning to be introduced in subsection 2.4.2 identifies all required a-types for a given fixed set of facet normals.

2.3.2 Mixed Volumes

Definition. Given an $\mathcal{S}$-representation with convex bodies S_i in n-dimensional space, the volume of any polytope $P(\boldsymbol{\lambda})$ with $\lambda_i \geq 0$ can be calculated by:

$$\mu^{\mathrm{volume}}(\boldsymbol{\lambda}) = \mu^{\mathrm{volume}}\left(\sum_i \lambda_i S_i\right) = \sum_{i_1,\ldots,i_n=1}^{n_S} \lambda_{i_1}\lambda_{i_2}\cdots\lambda_{i_n}\, \widetilde{V}_{i_1,\ldots,i_n}, \tag{2.32}$$

where the coefficients $\widetilde{V}_{i_1,\ldots,i_n}$ are denoted as mixed volumes that are functions of the corresponding structuring elements: $\widetilde{V}_{i_1,\ldots,i_n} = \widetilde{V}\left(S_{i_1}, \ldots, S_{i_n}\right)$ [Schneider, 2008, theorem 5.1.6]. Mixed volumes in this work are only applied in the context of a fixed set of convex bodies S_i so that the notation of $\widetilde{V}_{i_1,\ldots,i_n}$ is sufficient. In the following discussion, the dimension will be fixed to $n = 3$ to simplify the notation. All arguments analogously apply for arbitrary dimensions, where in particular the lower dimensions $n = 2$ or $n = 1$ are of interest.

Properties. Equation 2.32 for polytopes in $\mathcal{S}$-representation is analogous to equation 2.31 for polytopes in $\mathcal{H}$-representation. However, the remarkable property of the mixed volumes is that they are independent of the values λ_i, whereas the coefficients of equation 2.31 depend on the a-type of the polytope. Additionally, some special mixed volumes are equal to typical measures like the surface area or the Feret diameter, which will be discussed in the appendix A.1.1.

Mixed volumes are symmetric and the order of the arguments does not change the value:

$$\widetilde{V}_{1,2,3} = \widetilde{V}_{1,3,2} = \widetilde{V}_{2,1,3} = \widetilde{V}_{2,3,1} = \widetilde{V}_{3,1,2} = \widetilde{V}_{3,2,1}. \tag{2.33}$$

In addition, any mixed volume $\widetilde{V}(S_i, S_i, S_i)$ equals the volume $\mu^{\text{volume}}(S_i)$, which makes it consistent to the monotony order of the volume functional: $\mu^{\text{volume}}(\lambda S) = \lambda^3\, \mu^{\text{volume}}(S)$.

Determining Mixed Volumes. Mixed volumes can be determined by calculating the volumes: $\mu^{\text{volume}}(S_i)$, $\mu^{\text{volume}}(S_i + S_j)$, $\mu^{\text{volume}}(S_i + 2\,S_j)$ and $\mu^{\text{volume}}(S_i + S_j + S_k)$. Any algorithm to calculate these volumes is suitable. As the mixed volumes are symmetric, only the index combinations $i \leq j \leq k$ are required. First, the mixed volumes for $i = j = k$ are readily available with:

$$\widetilde{V}_{i,i,i} = \mu^{\text{volume}}\left(S_i\right). \tag{2.34}$$

Secondly, from equation 2.32 it follows:

$$\mu^{\text{volume}}(S_i + S_j) = \widetilde{V}_{i,i,i} + \widetilde{V}_{j,j,j} + 3\,\widetilde{V}_{i,i,j} + 3\,\widetilde{V}_{i,j,j} \tag{2.35}$$

$$\mu^{\text{volume}}(S_i + 2\,S_j) = \widetilde{V}_{i,i,i} + 8\,\widetilde{V}_{j,j,j} + 6\,\widetilde{V}_{i,i,j} + 12\,\widetilde{V}_{i,j,j}, \tag{2.36}$$

which results in the mixed volumes for $i = j < k$ and $i < j = k$ after elimination of $V_{i,i,j}$ and $V_{i,j,j}$:

$$\widetilde{V}_{i,j,j} = \frac{1}{6}\widetilde{V}_{i,i,i} - \widetilde{V}_{j,j,j} - \frac{1}{3}\,\mu^{\text{volume}}(S_i + S_j) + \frac{1}{6}\mu^{\text{volume}}(S_i + 2S_j) \tag{2.37}$$

$$\widetilde{V}_{i,i,j} = -\frac{1}{2}\widetilde{V}_{i,i,i} + \frac{2}{3}\widetilde{V}_{j,j,j} + \frac{2}{3}\mu^{\text{volume}}(S_i + S_j) - \frac{1}{6}\mu^{\text{volume}}(S_i + 2S_j). \tag{2.38}$$

And finally, from equation 2.32 it follows:

$$\begin{aligned}\mu^{\text{volume}}(S_i + S_j + S_k) = {} & \widetilde{V}_{i,i,i} + \widetilde{V}_{j,j,j} + \widetilde{V}_{k,k,k} \\ & +3\,\widetilde{V}_{i,i,j} + 3\,\widetilde{V}_{i,i,k} + 3\,\widetilde{V}_{i,j,j} + 3\,\widetilde{V}_{i,k,k} \\ & +3\,\widetilde{V}_{j,j,k} + 3\,\widetilde{V}_{j,k,k} + 6\,\widetilde{V}_{i,j,k},\end{aligned} \tag{2.39}$$

which results in the mixed volumes for $i < j < k$:

$$\begin{aligned}\widetilde{V}_{i,j,k} = {} & \frac{1}{6}\Big(\mu^{\text{volume}}(S_i + S_j + S_k) - \widetilde{V}_{i,i,i} - \widetilde{V}_{j,j,j} - \widetilde{V}_{k,k,k} \\ & -3\,\widetilde{V}_{i,i,j} - 3\,\widetilde{V}_{i,i,k} - 3\,\widetilde{V}_{i,j,j} - 3\,\widetilde{V}_{i,k,k} - 3\,\widetilde{V}_{j,j,k} - 3\,\widetilde{V}_{j,k,k}\Big).\end{aligned} \tag{2.40}$$

All required coefficients $\widetilde{V}_{i,j,k}$ with $i \leq j \leq k$ are determined with equations 2.34, 2.37, 2.38 and 2.40 and all remaining coefficients follow by symmetry of the mixed volumes.

Outlook. This subsection defines mixed volumes and provides the most important tools for their application. Section 2.3.3 and section A.1.1 will demonstrate several relations to frequently used polytope measures. The application of mixed volumes, however, is not demonstrated until section 3.3.5 in the context of the geometric state space of crystals.

2.3.3 Polytope Measures calculated by Mixed Volumes

Introduction. This section summarizes how different measures of a polytope in $\mathcal{S}$-representation can be calculated by mixed volumes (equation 2.32). These measures are, on top of the polytope volume itself: surface area, projection area, area of specific facets, Feret diameter, length of specific edges and mean diameter. The measures are grouped according to the theorem that is required to deduce their relation to mixed volumes. For a deeper understanding the reader is also referred to appendix A.1.1 which explains how the aforementioned measures are directly represented by special mixed volumes.

Support sets: specific edge length and facet area. The facets and edges of a polytope are denoted by the support sets $F(P(\mathbf{h}), \mathbf{u})$ in suitable directions $\mathbf{u} \in S^{(n-1)}$. For any support set of a polytope in $\mathcal{S}$-representation, it holds: $F(\sum_i S_i, \mathbf{u}) = \sum_i F(S_i, \mathbf{u})$ from equation 2.28 in section 2.2.3. The facet area is then the volume of the 2-dimensional support set measured in the corresponding 2-dimensional subspace:

$$\mu^{\text{facet area}}\left(\sum_i \lambda_i S_i, \mathbf{u}\right) = \sum_{i,j} \lambda_i \lambda_j \tilde{V}\left(F(S_i, \mathbf{u}), F(S_j, \mathbf{u})\right). \tag{2.41}$$

The mixed volumes in this equation can be determined analogously to the 3-dimensional mixed volumes. The edge lengths are calculated by 1-dimensional mixed volumes:

$$\mu^{\text{edge length}}\left(\sum_i \lambda_i S_i, \mathbf{u}\right) = \sum_i \lambda_i \tilde{V}_i\left(F(S_i, \mathbf{u})\right), \tag{2.42}$$

where $\tilde{V}_i\left(F(S_i, \mathbf{u})\right)$ are the lengths of the corresponding edges for the structuring elements S_i.

Note that the vectors $\mathbf{u}$ that might determine a 2-dimensional support set are known *a priori* by the facet normal vectors $\mathbf{a}_i$ for the application to crystals. Edge length measures are not applied in this work. A suitable vector $\mathbf{u}$ for an edge between two facets with the normal vectors $\mathbf{a}_i$ and $\mathbf{a}_j$ might be denoted by $\mathbf{u} = \frac{1}{2}\mathbf{a}_i + \frac{1}{2}\mathbf{a}_j$. However, equation 2.42 measures the 1-dimensional volume of the corresponding support set, which is not necessarily the intended edge. It might even be a 2-dimensional support set, if a corresponding facet normal exists.

Projections: projection area and Feret diameter. A projection is a linear operator which maps any original point into some, usually lower dimensional, subspace. The projection is viewed directly as a mapping from the originally n-dimensional space to the lower dimensional space of dimension $m \leq n$. For a polytope P and a $m \times n$ mapping matrix M, this can be expressed by:

$$\mathbf{M}\,P = \{\mathbf{M}\,\mathbf{x} \mid \mathbf{x} \in P\}. \tag{2.43}$$

From the definition of the Minkowski addition (equation 2.21) and the distributivity for matrix multiplication, it follows:

$$\mathbf{M}\left(\sum_i \lambda_i S_i\right) = \sum_i \lambda_i \left(\mathbf{M}\,S_i\right). \tag{2.44}$$

This equation is analogous to equation 2.28 for support sets. The plane for the projection is given by its normal vector $\mathbf{u}$ and the projected polytope by $\mathbf{M_u}\,P$ for which $\mathbf{M_u}$ is a suitable 2×3 matrix that maps to the 2-dimensional subspace. The projection area is then denoted and given by:

$$\mu^{\text{projection area}}\left(\sum_i \lambda_i S_i,\ \mathbf{u}\right) = \sum_{i,j} \lambda_i \lambda_j \tilde{V}\left(\mathbf{M_u}\,S_i,\ \mathbf{M_u}\,S_j\right). \tag{2.45}$$

The Feret diameter is the distance between two parallel supporting hyperplanes that is: $h\left(\sum_i \lambda_i S_i,\ \mathbf{u}\right) + h\left(\sum_i \lambda_i S_i,\ -\mathbf{u}\right)$, where $\mathbf{u}$ is the facet normal of one of the supporting hyperplanes. In the notation from equation 2.43 and 2.44 the mapping matrix is $\mathbf{M} = \mathbf{u}^T$. The Feret diameter is then denoted and given by:

$$\mu^{\text{Feret diameter}}\left(\sum_i \lambda_i S_i,\ \mathbf{u}\right) = \sum_i \lambda_i \tilde{V}\left(\mathbf{u}^T S_i\right). \tag{2.46}$$

Quermasses. The following paragraph will introduce additional measures for which the projection area and the Feret diameter are used in their adequate mathematical terminology. The Feret diameter is denoted by the width of a polytope in direction $\mathbf{u}$ and both, the projection area and the Feret diameter, are denoted as quermasses that apply for general orthogonal projections of n-dimensional bodies into m-dimensional subspaces.

Minkowski functionals: surface area and mean width. The expectation value of a quermass, which is the Feret diameter or the projection area in this case, according to directions $\mathbf{u} \in S^{n-1}$ is found in mathematical literature by: 'quermassintegrals', 'Minkowski functionals' or 'intrinsic volumes'. All these measures are related with some constant factor [Schneider, 2008, section 4.2]. Detailed descriptions are skipped for the sake of brevity but can be found in textbooks about integral geometry [Santaló, 1976, section 13.1] or convex geometry or [Bonnesen and Fenchel, 1934, Schneider, 2008, section 13.1]. The relation between the surface area and the expectation value of the orthogonal projections onto a plane is more commonly known as the Cauchy-theorem and the ratio of the surface area divided by the expectation value is 4.

In summary, the surface area is denoted and given by:

$$\mu^{\text{surface}}\left(\sum_i \lambda_i S_i\right) = \sum_{i,j} \lambda_i \lambda_j \tilde{V}_{i,j} \tag{2.47}$$

and the mean width by:

$$\mu^{\text{mean width}}\left(\sum_i \lambda_i S_i\right) = \sum_i \lambda_i \tilde{V}_i. \tag{2.48}$$

The surface area of a polytope can be calculated by summing up all the specific facet areas. Calculating the mean width requires much more detail. It is also of central importance for the modeling of rounded shapes in chapter 5 so that this discussion follows in subsection 2.3.4, separately.

2.3.4 Mean Width

Introduction. The mean width was already introduced in subsection 2.3.3. It can be written, as:

$$\mu^{\text{mean width}}(P) = \frac{\int_{\mathbf{u}\in S^{n-1}} h(P,\,\mathbf{u}) + h(P,\,-\mathbf{u})\,\mathrm{d}\mathbf{u}}{\int_{\mathbf{u}\in S^{n-1}} \mathrm{d}\mathbf{u}},$$

even though a more accurate notation is used in mathematical textbooks [Schneider, 2008, equation 1.7.2]. The term $h(P,\mathbf{u}) + h(P,-\mathbf{u})$ is the width of P with respect to the direction $\mathbf{u} \in S^{n-1}$. The width equals the Feret diameter in the context of particle measurements so that the mean width equals the mean of the Feret diameter. This section will introduce two more alternative definitions. The first one provides an equation to calculate the mean width and the second definitions provides a visually expressive interpretation for the mean width of 3-dimensional polytopes.

Relation to other measures. The mean width of a convex body $P \in \mathbb{R}^3$ is related to the 2^{nd} Minkowski functional $W_2(P)$, the mixed volume $V(P,B,B)$ with B being the unit ball and the intrinsic volume $V_1(P)$ [Schneider, 2008, page 210 and equation 5.3.8]:

$$\mu^{\text{mean width}}(P) = \frac{3}{2\pi} W_2(P) = \frac{3}{2\pi} V(P,B,B) = V_1(P). \tag{2.49}$$

The mean width is additionally related to the integral of the mean curvature, but the curvature of a non-smooth body like a polytope cannot be defined.

In 2-dimensions, the mean width of a disc is its diameter and π-times the mean width equals the circumference of the disc. This relation between the mean width and the circumference, surprisingly, holds true for arbitrary 2-dimensional convex bodies. For 3-dimensional polytopes, a similar relation between the mean width and the sum of the edge lengths will become evident in the following.

Definition by intrinsic volumes. The relation to intrinsic volumes gives explicitly [Schneider, 2008, equation 4.2.30]:

$$\mu^{\text{mean width}} = \sum_i \gamma(F_i)\,\mu^{\text{length}}(F_i) \tag{2.50}$$

where the summation is taken over all edges F_i of the polytope. The term $\gamma(F_i)$ is the external angle of the corresponding edge that is proportional to the angle between the normals of the facets that create the edge. The term $\mu^{\text{length}}(F_i)$ is the length of the edge.

Definition by mixed volumes. While the mean width could already be calculated based on equation 2.50, the interpretation of this equation is more intuitive by the following result for mixed volumes (equation 2.32):

$$\mu^{\text{volume}}(P + \lambda B) = \tilde{V}(P, P, P) + 3\lambda\,\tilde{V}(P, P, B) + 3\lambda^2\,\tilde{V}(P, B, B) + \lambda^3\,\tilde{V}(B, B, B) \quad (2.51)$$

which is also known as the Steiner formula if the mixed volumes are replaced by the corresponding Minkowski functionals [Schneider, 2008, page 210]. The first term $\tilde{V}(P, P, P)$ and the last term $\tilde{V}(B, B, B)$ are apparently the volumes of the body P and the unit ball B, respectively. The term $3\,\tilde{V}(P, P, B)$ is also well known as the surface area of the body P. Therefore, the term $3\tilde{V}(P, B, B)$ that relates to the mean width is so far the only term without a clear visual interpretation.

Alternative interpretation of the mean width. The body $P + \lambda B$ can be decomposed to depict the four terms from equation 2.51 to visualize the meaning of equation 2.50. Figure 2.7 shows the wireframe of the body $P + \lambda B$ in the outer left. The initial volume of the body P is drawn in the outer left figure. The volume of the ball λB is obtained by the parts visualized in the outer right figure. The vertices of a convex polytope will always create a full ball λB, if $P + \lambda B$ is decomposed in this way. The inner left part shows the volumes ($3\lambda\,\tilde{V}(P, P, B)$) that are created by multiplying the surface area of each facet by λ. What remains for the inner right figure are volumes that are created by multiplying the length of each edge ($\mu^{\text{length}}(F_i)$) with the external angle $\gamma(F_i)$. Finally, the external angle can be interpreted in this case as the angle between the normal vectors of the adjacent facets divided by the full angle times the area of the unit disc (for $\lambda = 1$).

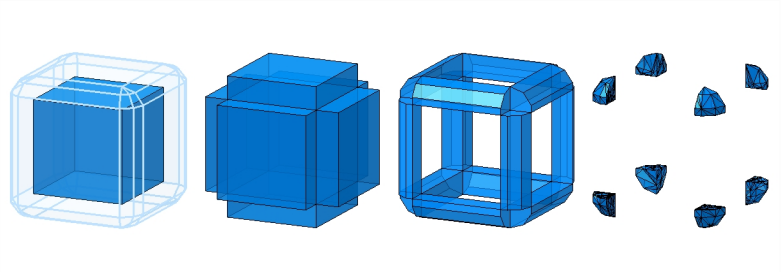

Figure 2.7.: Decomposition of the body $P + \lambda B$ (wireframe in the outer left figure) into parts that correspond to equation 2.51: the volume of the body P (outer left), the volume that equals $\lambda V(P, P, B)$ (left), the volume that equals $\lambda^2 V(P, B, B)$ (right) and the volume of $\lambda^3 B$ (outer right)

2.4 Polytope Surface Structure

2.4.1 Preliminaries on Minkowski Decomposition

Motivation. The concept of mixed volumes provides a promising alternative to calculate physically meaningful measures. However, a crystal is naturally given in $\mathcal{H}$-representation with a fixed set of facet normals while mixed volumes assume an $\mathcal{S}$-representation. The theory that is introduced in this subsection creates an initial link between these two representations. It defines a reasonable set of structuring elements for a given polytope. The corresponding algorithm that is outlined in the following subsection is used in chapter 3 to identify a minimal set of structuring elements for a given set of crystal facets.

Delimitation of focus. The ideas and concepts, discussed in this subsection, originate from a problem that is called 'Minkowski decomposition' in mathematical literature. Therefore, this and the following subsection are denoted by this term. The required terminology and ideas to solve this problem are, however, much more useful than its solution.

Relation on polytopes. Given two polytopes P and Q in n-dimensional space, $P \leq Q$ is written if it holds:

$$\dim F(P, \mathbf{u}) \leq \dim F(Q, \mathbf{u}) \tag{2.52}$$

for the support sets $F(\cdot, \mathbf{u})$ in any direction $\mathbf{u} \in S^{(n-1)}$ [Meyer, 1974] while $\dim$ denotes the dimension of the support set. This implies for 3-dimensional polytopes that Q has a facet in all directions $\mathbf{u}$ for which P has a facet. Additionally, Q has an edge or a facet in all directions $\mathbf{u}$ for which P has an edge.

If the outer left polytope in figure 2.8 is Q, any polytope on its right could be inserted for P so that $P \leq Q$ holds. In fact, any polytope $P(\boldsymbol{\lambda})$ with $\lambda \geq 0$ can be used where $P(\boldsymbol{\lambda})$ equals $\sum_i \lambda_i P_i$ and P_1, P_2 and P_3 being the three polytopes in the right of figure 2.8.

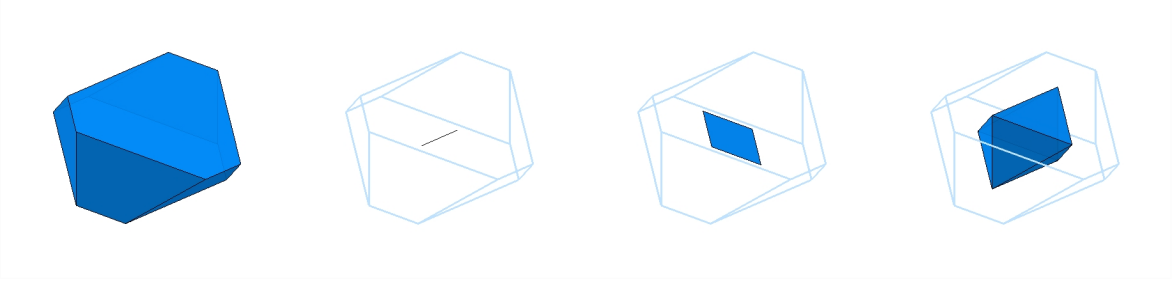

Figure 2.8.: A polytope (outer left) as the Minkowski sum of the three polytopes on the right; the summands are viewed together with the wireframe of the sum

Possible summands. Minkowski addition conveys information on the surface structure from its summands to the sum (see equation 2.28) so that it is evident that $P \leq Q$ implies that P is a summand of Q [Shephard, 1963, Meyer, 1974]. Here, any polytope P is called a summand of the polytope Q if a third polytope R exists, with: $\lambda Q = P + R$ and $\lambda \geq 0$. The relation $P \leq Q$ is also sufficient for P being a summand of Q which subsequently implies that any summand can be represented by the same set of facet normals in a $\mathcal{H}$-representation.

From the example above, any polytope $P(\boldsymbol{\lambda})$ with $\lambda \geq 0$ is a summand of Q.

Indecomposability. A polytope Q is called indecomposable, if and only if there are exclusive decompositions of the form $Q = P + R$ with $P = \lambda Q + \{\mathbf{x}\}$ and $R = (1 - \lambda)Q - \{\mathbf{x}\}$ for $0 \leq \lambda \leq 1$ and $\mathbf{x} \in \mathbb{R}^n$. This means that for any possible decomposition into summands there exist only summands that have exactly the shape of Q but different size and position.

All three polytopes in the right of figure 2.8 are indecomposable.

Minkowski decomposition. Given a polytope Q, the decomposition of Q into indecomposable polytopes P_i with $Q = \sum_i P_i$ is called *Minkowski decomposition*. A Minkowski decomposition of a shape can be considered analogous to the prime factor decomposition of a number. However, a Minkowski decomposition is not unique.

Figure 2.8 shows a polytope on the left that was already introduced as Q. The three polytopes in the right are the three indecomposable polytopes of a Minkowski decomposition. While the surface structure of the original polytope is relatively complex, the indecomposable polytopes have a simple surface structure with only a few faces, edges and vertices.

a-type. If it holds $P \leq Q$ and $Q \leq P$ for any two polytopes P and Q, they are called strongly isomorphic [Meyer, 1974]. Strongly isomorphic is formerly denoted by 'analogous' in mathematical literature and, thus, a class of strongly isomorphic polytopes is called an a-type [Schneider, 2008, page 100]. The a-type for a 3-dimensional polytope represents the combination of faces, edges and vertices that are present so that the term a-type matches the term morphology in the context of crystal shapes.

The a-type represents an equivalence class for polytopes that is denoted by $[P]$ for a representative polytope P. Consequently, the discussion on single polytopes can be transferred to a discussion on equivalence classes (*e.g.* writing '$[P] \leq [Q]$' instead of '$P \leq Q$ with $P \in [P]$ and $Q \in [Q]$').

Conclusion. The discussion above shows that complex issues concerning the morphology of crystals can be expressed by a simple relation: $P \leq Q$. Additionally, it is demonstrated that using $P \leq Q$, speaking about Minkowski summands or discussing crystal morphology are strongly related to each other.

With the introduced terminology it is now possible to identify a set of structuring elements to rewrite all polytopes $P \leq Q$ as an $\mathcal{S}$-representation with a fixed set of structuring elements. First, all indecomposable polytopes $P_i \leq Q$ must be contained in that set because they cannot be expressed as the Minkowski sum of other polytopes. Secondly, any polytope P' that does not fulfill $P' \leq Q$ cannot be a summand of any polytope $P \leq Q$ and, hence, can be omitted. Finally, adding any other polytope $P' \leq Q$ to the structuring elements is superfluous because it can be substituted in an $\mathcal{S}$-representation by an expression of the indecomposable polytopes. Hence, exactly all indecomposable polytopes $P_i \leq Q$ are required as structuring element to express any polytope $P \leq Q$ as an $\mathcal{S}$-representation.

2.4.2 Minkowski Decomposition in h-space

Overview. In this subsection, the implications of the relation $P \leq Q$ are viewed in the space of $\mathbf{h}$-vectors for a $\mathcal{H}$-representation that is suitable to represent Q. The domains for the input a-type (all vectors $\mathbf{h}$ with $P(\mathbf{h}) \in [Q]$) and the domain of all possible

summands (all vectors $\mathbf{h}$ with $P(\mathbf{h}) \leq Q$) are presented. Defining these regions goes along with outlining elements of an algorithm to calculate these domains and, in particular, the indecomposable polytopes that are summands of Q. Therefore, remarks on this algorithm and computational issues form the second part of this subsection. Note, however, that the focus of this subsection is still the link between $\mathcal{H}$- and $\mathcal{S}$-representation to exploit the measure calculation based on mixed volumes which is commented on in the last paragraph.

Normalized polytope positions. It is beneficial to define and fix the position of each considered polytope because it represents superfluous information. Moving the polytope P or Q does not change the relation $P \leq Q$. The appearing complexity when the polytope position is not normalized is demonstrated in the subsequent paragraph on indecomposable polytopes.

Regardless of the definition, the chosen polytope position should be invariant to Minkowski addition so that the sum is centered if all summands are centered. This is obtained whenever the polytope position changes linearly dependent on the components of the $\mathbf{h}$-vector and follows directly from the definition of Minkowski addition. Assuming the polytope Q is centered, it is then suitable to demand that all summands are centered. This introduces n linear constraints to the corresponding $\mathbf{h}$-vectors so that any entity in $\mathbf{h}$-space will be embedded in an $(n_\mathrm{H} - n)$-dimensional subspace that is given by the definition of the polytope position.

Meyer [1974] centers polytopes at the Steiner point. For an n-dimensional polytope P, this point is introduced as the expectation value of $(h(P, \mathbf{u}) \cdot \mathbf{u})$ with $\mathbf{u}$ sampled uniformly from the unit hypersphere S^{n-1} [Grünbaum, 2003, section 14.5]. The Steiner point is invariant to Minkowski addition. However, while this point is useful for mathematical proofs, it is hard to compute so that the normalization of polytope positions will be adopted in chapter 3.

Indecomposable polytopes. If a summand P of Q is indecomposable, it holds that any $P' \in [P]$ equals $P' = \lambda P$ with $\lambda \geq 0$. Since $\lambda P(\mathbf{h}) = P(\lambda \mathbf{h})$ holds, indecomposable polytopes are represented by single rays in $\mathbf{h}$-space. This does only hold if normalized polytope positions are assumed. If the positions are not fixed, indecomposable polytopes would be represented by some n-dimensional subdomains which would drastically increase the complexity of the remaining discussion.

Linear equality constraints. If a component of the $\mathbf{h}$-vector for an indecomposable polytope is changed slightly so that the a-type changes, at least one edge (ridge for n-dimensional polytopes) emerges. This edge (ridge) equals the connection between two extreme points and it follows a discussion on how this edge (ridge) is defined by $(n+1)$ facets. Restricting the corresponding edge length to 0 then creates an equality constraint to the vector $\mathbf{h}$.

Each extreme point can be defined by the intersection of n facets of the polytope. The edge (ridge) is embedded in a 1-dimensional subspace given by the intersection of $(n-1)$ facets. Hence, both extreme points of the edge can be defined by $(n+1)$ facets: two sets of n facets for each extreme point while the facet sets can be chosen to have the $(n-1)$ facets of the edge (ridge) in common. The length of this edge changes linearly in the components of the $\mathbf{h}$-vector (see page 21), it is defined by $(n+1)$ facets and it must remain 0 for indecomposable polytopes in order to add no new edge (ridge). This

represents a linear constraint:

$$\mathbf{a}^T\mathbf{h} = 0 \tag{2.53}$$

with at most $(n+1)$ non-zero elements in $\mathbf{a}$. How such constraints are constructed is explained in [Meyer, 1974].

In conclusion, any four facets of Q that intersect in a common extreme point create a corresponding linear constraint to the vector $\mathbf{h}$. Any polytope $P(\mathbf{h}) \leq Q$ must fulfill this equation since any additional edge in $P(\mathbf{h})$ violates $P(\mathbf{h}) \leq Q$.

Linear inequality constraints. The term $\mathbf{a}^T\mathbf{h}$ is already introduced to represent the length of edges, given that the sign is chosen accordingly. Consequently, it must hold:

$$-\mathbf{a}^T\mathbf{h} < 0 \tag{2.54}$$

for existing edges of Q whenever $\mathbf{h}$ represents a polytope of the same a-type: $P(\mathbf{h}) \in [Q]$.

Domains. With equations 2.53 and inequalities 2.54, the region of $\mathbf{h}$-vectors is defined for which $P(\mathbf{h}) \in [Q]$ holds. If an edge length becomes 0 by inequation 2.54, the a-type changes while $P(\mathbf{h}) \leq Q$ is retained. If the violated inequation is altered to an equality constraint, the vectors $\mathbf{h}'$ fulfilling this new set of equations and inequations will represent polytopes $P(\mathbf{h}') \leq P(\mathbf{h}) \leq Q$. This indicates a procedure that can be repeated several times. Each time, one or more edges disappear and the a-type changes. This process ends, when the range of $\mathbf{h}'$ represents a 1-dimensional ray and, hence, an indecomposable polytope. This implies that the domain of all summands $P(\mathbf{h}) \leq Q$ is represented by the weak form ($\leq$) of the strict inequalities 2.54 and the equations 2.53. This domain is a convex cone and every indecomposable polytope is given as an extreme ray of that cone. While this domain defines the region of possible summands, it does simultaneously define the region in $\mathbf{h}$-space that is covered by the $\mathcal{S}$-representation which is comprised of the corresponding indecomposable polytopes.

Algorithm. It can be concluded, that finding the indecomposable polytopes P_i or, respectively, the extreme rays of the cone of all summands for Q is sufficient to resolve any question connected to Minkoswki summands $P \leq Q$. An algorithm that defines the range of all summands $P \leq Q$ is given by Meyer [1974] or Mount and Silverman [1989] for n-dimensional polytopes. The work of Meyer [1974] is recommended for a proof of this approach and is the primary source for the introduced notation. Alternatively, a similar algorithm is described by Borchert [2012] for the morphology of 3-dimensional crystals. This source is recommended for an alternative approach to the derivation that does not include Minkowski addition. It uses a different notation and handles morphology in a less generalized manner.

Both algorithms, if applied for 3-dimensional polytopes, exhaustively iterate through all quadruples of facets. The equations 2.53 are constructed wherever four facets have an extreme point in common. The inequations 2.54 are constructed for every edge of the input polytope Q.

Evaluation. The cone of summands is embedded in some lower dimensional subspace of the $\mathbf{h}$-space. Computations on such degenerated polyhedra are not beneficial so that the cone of summands is mapped in the proper linear subspace. A suitable mapping can be computed by singular value decomposition of the matrix that contains all constraints. The resulting rays of the full dimensional cone are calculated by the double description

method [Fukuda, 1996, 2008]. After mapping the extreme rays back to the original h-space, the indecomposable polytopes $P_i(\mathbf{h}_i)$ are obtained.

The original problem statement for Minkowski decomposition is solved for any linear combination $\lambda_i \geq 0$ with: $\mathbf{h} = \sum \lambda_i \mathbf{h}_i$ so that the summands of the Minkowski decomposition are given by $P(\lambda_i \mathbf{h}_i)$.

Application. It is provided that, whenever all polytopes $P(\mathbf{h}_i)$ are summands of the polytope $P\left(\sum_i \lambda_i \mathbf{h}_i\right)$ it holds:

$$P\left(\sum_i \lambda_i \mathbf{h}_i\right) = \sum_i \lambda_i P(\mathbf{h}_i) \tag{2.55}$$

for $\lambda_i \geq 0$. This can be followed from Schneider [2008, theorem 1.7.5] or the definition of the Minkowski addition (see equation 2.21). Equation 2.55 implies that the polytope $P(\mathbf{h}) = \sum_i \lambda_i P(\mathbf{h}_i)$ can be calculated without explicitly performing a Minkowski addition by:

$$\mathbf{h} = \sum_i \lambda_i \mathbf{h}_i \tag{2.56}$$

in $\mathcal{H}$-representation. Hence, equation 2.56 provides a direct relation between the $\mathcal{S}$-representation (the coefficients λ_i) and the $\mathcal{H}$-representation (the vector of facet distances $\mathbf{h}$).

The suggested $\mathcal{S}$-representation is comprised of the indecomposable polytopes $P(\mathbf{h}_i) \leq Q$ as structuring elements. The region that represents all summands in h-space represents, hence, also the region of all polytopes that are covered by this $\mathcal{S}$-representation and for which equation 2.56 can be applied. More discussion on this follows in chapter 3.

2.4.3 Validity of Support Hyperplane Data

Motivation. The algorithm that is introduced in this subsection is applied for the crystal state space that is not yet defined. Hence, the following describes the motivation that is used in the original sources [Karl et al., 1995, Mount and Silverman, 1989].

A support function $h(B, \mathbf{u})$ defines any convex body B when $\mathbf{u}$ is taken from the unit hypersphere $S^{(n-1)}$ (see page 11). Hence, given a real 3-dimensional convex body, it is natural to choose an arbitrary origin and measure a discrete number of support values $h(B, \mathbf{a}_i)$ to describe this body. This description yields a polytope $P_{\mathbf{A}}(\mathbf{h})$ for which the facet normal matrix $\mathbf{A}$ contains the measurement directions and the vector of facet distances contains the support value measurements. However, due to measurement errors, the set of measured support values $h(B, \mathbf{a}_i)$ for some fixed measurement directions $\mathbf{a}_i$ is in general not consistent with the support measures of the identified polytope $P(\mathbf{h})$: $h(P(\mathbf{h}), \mathbf{a}_i)$. Related problems are entitled 'validity of support hyperplane data' or 'convex set reconstruction' in mathematical literature. They include, in particular, whether a measured vector $\mathbf{h}$ is consistent, or not. This problem statement is equivalent to important questions in crystallization modeling. Given a fixed facet normal matrix $\mathbf{A}$: for which vectors of facet distances $\mathbf{h}$ are all facets present? Or: which facets appear for the polytope $P(\mathbf{h})$? These questions are answered in this subsection.

Mathematical literature. An algorithm for the problem above is presented by Karl et al. [1995]. It comprises a generalization for n-dimensional polytopes with some improvements to prior work by Prince and Willsky [1990, 1991]. These improvements were implemented but the time savings were found insufficient compared to the added complexity of the algorithm. Karl et al. [1995] assume that the polytope that is described by the support measures is not empty. This is, in general, not true for the purpose in this work so that cases must be taken into account that were neglected in the original paper. The algorithm is outlined in the following with the main arguments and their geometrical interpretation. It is described for 3-dimensional polytopes, even though it is available for n-dimensional objects.

Skeleton of the Algorithm. The algorithm investigates all ordered sets of 4 facet normals. The first three facet normals are denoted by $\mathbf{a}'_1$, $\mathbf{a}'_2$ and $\mathbf{a}'_3$ and constitute a possible vertex for some polytopes $P(\mathbf{h})$. Figure 2.9 illustrates four examples that are used in the subsequent discussion. The blue facets of the lower tetrahedron are created by the facet normals $\mathbf{a}'_1$, $\mathbf{a}'_2$ and $\mathbf{a}'_3$ while the upper tip of the wireframe represents the vertex. The fourth dark blue facet that belongs to the normal $\mathbf{a}'_4$ cuts the tetrahedron in different angles. Furthermore, the positive and a negative cones are considered which are formed by the normals $\mathbf{a}'_1$, $\mathbf{a}'_2$ and $\mathbf{a}'_3$. The positive cone is the positive hull of the normals interpreted as points in 3-dimensional space. It is drawn in light blue in figure 2.9. The negative cone uses the points $-\mathbf{a}'_1$, $-\mathbf{a}'_2$ and $-\mathbf{a}'_3$, instead, and is not drawn.

Since only facet normals with $\operatorname{rank}(\mathbf{a}'_1, \mathbf{a}'_2, \mathbf{a}'_3) = 3$ intersect in a vertex, combinations of facet normals with $\operatorname{rank}(\mathbf{a}'_1, \mathbf{a}'_2, \mathbf{a}'_3) < 3$ can be ignored. The remainder of the analysis is based on three cases: the facet normal $\mathbf{a}'_4$ resides in the positive cone; it resides in the negative cone[9]; or it resides neither in the positive nor in the negative cone. According to Karl et al. [1995, section 3.1] only the first two cases must be considered.

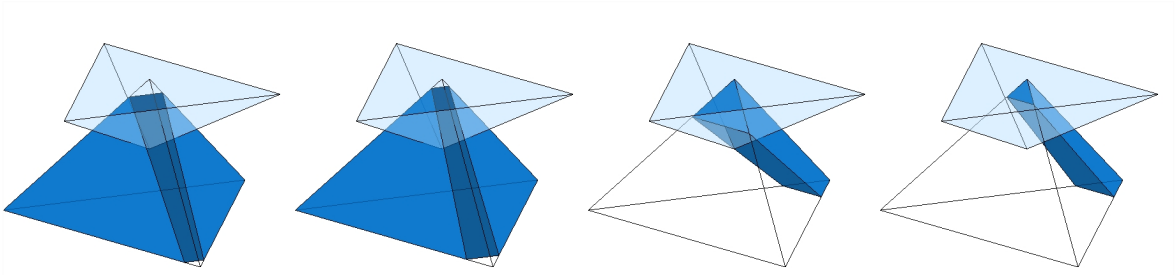

Figure 2.9.: Four examples how a vertex (top vertex of the lower tetraeder which is drawn as a wireframe) can be cut by a facet (dark blue); the positive cone formed by the facet normals defining the vertex is drawn in light blue

Positive cone. If the facet normal $\mathbf{a}'_4$ is in the positive cone, this facet might disappear at the vertex $\mathbf{v}'$ given by $\mathbf{a}'_1$, $\mathbf{a}'_2$, $\mathbf{a}'_3$ and the corresponding facet distances h'_1, h'_2 and h'_3:

$$\left(\mathbf{a}'_1, \mathbf{a}'_2, \mathbf{a}'_3\right)^T \mathbf{v}' \quad = \quad \left(h'_1, h'_2, h'_3\right)^T . \tag{2.57}$$

[9]The negative cone is not drawn in figure 2.9 but the statement '$\mathbf{a}'_4$ is in the negative cone' is equivalent to the statement '$-\mathbf{a}'_4$ is in the positive cone'.

The outer left example in figure 2.9 shows a facet normal $\mathbf{a}'_4$ that is in the positive cone. The corresponding facet would disappear at the vertex $\mathbf{v}'$. The inner left example in the same figure shows a facet normal $\mathbf{a}'_4$ that is just outside the positive cone. The corresponding facet would disappear at some other vertex. It follows that the facet corresponding to $\mathbf{a}'_4$ can only be present if the following inequation holds:

$$\left\langle \mathbf{v}', \mathbf{a}'_4 \right\rangle > h'_4. \tag{2.58}$$

Considering all (ordered) sets of 4 facets, several such inequations are generated for the same facet so that all of them must be fulfilled for the existence of this facet. If equality holds in one or more cases of these equations, the face just disappeared and the support measure $h(P(\mathbf{h}),\, \mathbf{a}'_4)$ is still consistent with h'_4. Concluding this case, every $\mathbf{h}$-vector must fulfill all inequations:

$$h'_4 - \left(h'_1, h'_2, h'_3\right) \left(\mathbf{a}'_1, \mathbf{a}'_2, \mathbf{a}'_3\right)^{-1} \mathbf{a}'_4 \leq 0. \tag{2.59}$$

so that it is consistent. Additionally, whenever one of the inequations for a specific facet is violated or equality holds, the corresponding facet is not present at the surface of the polytope $P(\mathbf{h})$.

Negative cone. If the facet normal $\mathbf{a}'_4$ is in the negative cone, the polytope $P(\mathbf{h})$ is empty for:

$$\left\langle \mathbf{v}', \mathbf{a}'_4 \right\rangle > h'_4. \tag{2.60}$$

The inner right example in figure 2.9 shows a facet normal $\mathbf{a}'_4$ that is in the negative cone. The whole polytope disappears at the vertex $\mathbf{v}'$ when h'_4 becomes too small.[10] The outer right example in the same figure shows a facet normal $\mathbf{a}'_4$ that is just outside the negative cone. The facet distance h'_4 might also become too small, but the polytope disappears at another vertex.

This case can be concluded, similar to the previous case. Every $\mathbf{h}$-vector must fulfill all inequations:

$$\left(h'_1, h'_2, h'_3\right) \left(\mathbf{a}'_1, \mathbf{a}'_2, \mathbf{a}'_3\right)^{-1} \mathbf{a}'_4 - h'_4 \leq 0. \tag{2.61}$$

to be consistent ($h_i = h(P(\mathbf{h}),\, \mathbf{a}_i)$ for all facets i). Whenever equality holds in one or more cases, no facet is present and $P(\mathbf{h})$ comprises a single point. Whenever any inequation 2.61 is violated, the polytope $P(\mathbf{h})$ is empty.

Final result. All inequations 2.59 and 2.61 represent a convex cone in $\mathcal{H}$-representation that can be summarized to:

$$\mathbf{A}'_{\mathrm{V}}\mathbf{h} \leq \mathbf{0}. \tag{2.62}$$

The vector of facet distances $\mathbf{h}$ is consistent if this inequation holds.

Additionally, the range of $\mathbf{h}$-vectors can be defined for which a specific facet is present. The inequations 2.61 for which the specific facet disappears and all inequations 2.61 are, therefore, summarized to:

$$\mathbf{A}'_{\mathrm{FV}(i)}\mathbf{h} < \mathbf{0}.$$

The facet with normal vector $\mathbf{a}_i$ is present when this inequation is fulfilled. As a special case, $i = 0$ summarizes only the inequations 2.61 so that $\mathbf{A}'_{\mathrm{FV}(0)}\mathbf{h} \leq \mathbf{0}$ implies that $P(\mathbf{h})$ is a non-empty polytope.

[10] Consider that $\mathbf{a}'_4$ is an outer normal vector so that the facet moves up for decreasing values h'_4.

3

The Geometric State Space Proper Representations

3.1 Introduction and Preliminaries

3.1.1 General Observations on Crystal Shape Modeling

Boundaries of validity. Let's assume a crystal representation uses only one size coordinate like the volume or the mean diameter. In such a case, there is typically only a lower bound of zero size and an upper bound does not exist. In contrast, using volume and surface area as size coordinates implies that the volume cannot become arbitrary large when its surface area is fixed. Another example is given by the growing facets of a crystal. In most cases, a growing facet has a maximum distance dependent on the other facet distances when they remain static. At this maximum distance, the facet disappears which is already discussed by Johnsen [1900]. While these upper bounds for chosen size coordinates can be avoided by chosing a sufficiently simple geometry (*e.g.* a cuboid), they easily arise in general cases such that they present an inherent complexity for crystal shape modeling. Since violating these upper bounds produces inconsistent combinations of shape coordinates, this concept is denoted by validity throughout this work.

Boundaries of zero size. Another difficulty applies to the lower bounds of the size coordinates at which crystals would disappear. When only one size coordinate is used, this point is typically zero. However, assume that the volume and the surface area are the size coordinates. In this case, the crystal disappears for any zero volume. This includes flat crystals with a non-zero surface area so that crystals can disappear at several positions in state space. Properly recognizing and applying these zero size boundaries adds another degree of complexity, in particular, for the formulation and solution of population balances.

Measure calculation. The driving force for the growth of crystals is supersaturation, the amount of dissolved substance above the saturation concentration. This amount is reduced with the growth of the crystals so that the total mass of the crystalline substance must be known in all time steps of a process simulation. Calculating the total volume of the simulated crystals is, therefore, crucial. Faceted crystals are polytopes for which the measure computation has been discussed in depth in subsection 2.3.1.

While the volume can be computed from a polynomial, the coefficients of that polynomial depend on the present morphology. This implies that the morphology of the crystal must be identified before the volume can be calculated. This remains true for various algorithms of volume computation except numerical integration[1] so that this problem does not depend on the selected crystal shape representation. Again, this complexity is avoided for a sufficiently simple geometry but presents, otherwise, an inherent difficulty of crystal shape modeling.

Outlook. For the crystal shape representation that is introduced in the next subsection, the validity and zero-size boundaries are defined and discussed in section 3.2. For the measure computation, it is possible to perform some calculations *a priori* so that simple operations remain for process simulations, optimization or identification. The corresponding algorithm is introduced in section 3.3 and identifies all required sets of coefficients for measure calculation including their region of applicability. Finally, the implemented algorithms are tested and the corresponding computational times are analyzed by a large scale case study in section 3.4.

3.1.2 Representation of Crystal Shape

Motivation. Using a $\mathcal{H}$-representation to represent faceted crystals is the most natural way since growth is represented directly by the time derivatives of the facet distances in $\mathbf{h}$ (see subsection 2.2.1). However, a lot of facets share the same growth rate due to symmetry, yielding $\mathbf{h}$-vectors with only a few differing entries. In consequence, the required independent variables for crystal shape modeling can be represented by a much lower dimensional vector $\mathbf{h}_\mathrm{C}$. The relation between the $\mathbf{h}$- and the $\mathbf{h}_\mathrm{C}$-vector are then modeled by a mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}$: $\mathbf{h} = \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}\,\mathbf{h}_\mathrm{C}$.

No assumption on the matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}$ is initially posed so that the analysis for applications of the resulting representations will indicate and define necessary properties. It turns out that these limitations are relatively tight which raises difficulties for more general representations that can occur for shape approximation or model reduction (see chapter 6 and appendix A.2).

$\mathcal{H}_\mathrm{C}$-representation. The general form of the shape representation that will be treated in this work uses any real-valued matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}$ to obtain the facet distances in the vector $\mathbf{h}$ from a lower dimensional vector $\mathbf{h}_\mathrm{C}$:

$$\mathbf{h} = \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}\mathbf{h}_\mathrm{C}. \tag{3.1}$$

This representation will be called constrained $\mathcal{H}$-representation or $\mathcal{H}_\mathrm{C}$-representation[2]. The dimension of the vector $\mathbf{h}_\mathrm{C}$ is given by n_C and the polytope $P(\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}\mathbf{h}_\mathrm{C})$ can be abbreviated by $P(\mathbf{h}_\mathrm{C})$. The matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}$ is called group mapping matrix because its unique rows group the facets into subsets that always share the same facet distance.

Note that the direct expression for a polytope is:

$$P_\mathbf{A}(\mathbf{h}_\mathrm{C}) = \{\mathbf{x} \mid \mathbf{A}\mathbf{x} \leq \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}\mathbf{h}_\mathrm{C}\},$$

which illustrates that a $\mathcal{H}_\mathrm{C}$-representation consists of the matrix of facet normals $\mathbf{A}$ and the mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}$ while the individual polytope is given by the vector $\mathbf{h}_\mathrm{C}$.

[1] Numerical integration is computationally inefficient for volume computation.

[2] While the term '$\mathcal{H}$-representation' is commonly used in mathematical literature, the terms 'constrained $\mathcal{H}$-representation' and '$\mathcal{H}_\mathrm{C}$-representation' are newly introduced.

The constraints to the vector $\mathbf{h}$ can be written explicitly by:

$$\mathbf{M}_{\mathrm{C}}\mathbf{h} = \mathbf{0}, \tag{3.2}$$

$$\mathbf{M}_{\mathrm{C}} = \left(\mathbf{I}_{n_{\mathrm{H}}} - \mathbf{M}_{\mathbf{h}_{\mathrm{C}}\mapsto\mathbf{h}}\mathbf{M}^{+}_{\mathbf{h}_{\mathrm{C}}\mapsto\mathbf{h}}\right) \tag{3.3}$$

where $\mathbf{I}_{n_{\mathrm{H}}}$ is the unity matrix of dimension $n_{\mathrm{H}} \times n_{\mathrm{H}}$ and $\mathbf{M}^{+}_{\mathbf{h}_{\mathrm{C}}\mapsto\mathbf{h}}$ is the pseudo inverse of $\mathbf{M}_{\mathbf{h}_{\mathrm{C}}\mapsto\mathbf{h}}$. The matrix $\mathbf{M}_{\mathrm{C}}$ and the mapping matrix $\mathbf{M}_{\mathbf{h}_{\mathrm{C}}\mapsto\mathbf{h}}$ can be used likewise to define a $\mathcal{H}_{\mathrm{C}}$-representation.

Remark on terminology. Groups of facets that are equivalent by symmetry are called a form in crystallographic terminology. They always have the same entries in a corresponding h-vector and they appear and disappear simultaneously on the crystal surface during growth [Borchardt-Ott, 2009]. If the $\mathcal{H}_{\mathrm{C}}$-representation is created according to crystal symmetry, the dimension n_{C} equals the number of modeled forms. However, since the shape representation above is more general and not necessarily describes a real crystal shape, the term facet group is usually applied instead of crystal form.

3.2 Validity and Proper Representations

3.2.1 Ill-conditioned Representations

Motivation. Information on the orientation or position of a polytope is irrelevant if only the particle shape or size is considered. While a $\mathcal{H}$-representation already fixes the orientation of the polytope in space, information on the position of a polytope is still conveyed such that several $\mathbf{h}$-vectors describe the same shape and size. The representation is, thus, not unique in the sense that multiple $\mathbf{h}$-vectors describe the same shape and size.

Overview. The ambiguity above can also apply for a $\mathcal{H}_\mathrm{C}$-representation so that the state space might comprise unnecessary degrees of freedom. It is shown in the second part of this subsection how to identify a $\mathcal{H}_\mathrm{C}$-representation that contains this ambiguity and how the representation can be altered to overcome this issue. In the first part, a more imminent ambiguity is discussed which is only connected to the group mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}$.

Group mapping matrix. Given a $\mathcal{H}_\mathrm{C}$-representation, it is expected that defining a group mapping matrix with n_C columns gives n_C degrees of freedom for the facet distances $\mathbf{h}$. This is apparently only true for:

$$\operatorname{rank}\left(\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}\right) = n_\mathrm{C}. \tag{3.4}$$

Any $\mathcal{H}_\mathrm{C}$-representation that does not fulfill equation 3.4 implies that a given vector $\mathbf{h} = \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}\mathbf{h}_\mathrm{C}$ can be expressed by multiple $\mathbf{h}_\mathrm{C}$-vectors so that the $\mathcal{H}_\mathrm{C}$-representation is ambiguous. This ambiguity can be resolved by removing columns of the matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}$ while the rank is maintained. Removing such columns reduces n_C while the set of achievable $\mathbf{h}$-vectors and, hence, the set of represented shapes is maintained.

Polytope position for $\mathcal{H}$-representations. If no constraint $\mathbf{M}_\mathrm{C}\mathbf{h} = \mathbf{0}$ is applied to a $\mathcal{H}$-representation, the polytope position could be defined as the intersection of three arbitrary but intersecting support hyperplanes (see page 11). Figure 3.1 illustrates this idea by a polytope drawn in blue and three intersecting support planes in light blue. The intersection is marked by a black spot and may, as shown there, reside outside of the polytope. If these three facets are fixed to a distance of zero, the polytope position equals the origin of the coordinate system. This implies that removing the position information from a $\mathcal{H}$-representation creates a $\mathcal{H}_\mathrm{C}$-representation with the dimension $n_\mathrm{C} = n_\mathrm{H} - 3$.

Fixed position for $\mathcal{H}_\mathrm{C}$-representations. Given an arbitrary $\mathcal{H}_\mathrm{C}$-representation, two polytopes $P(\mathbf{h}_\mathrm{C})$ and $P(\mathbf{h}'_\mathrm{C})$ might have the same shape and size but different positions with $\mathbf{h}_\mathrm{C} \neq \mathbf{h}'_\mathrm{C}$. This ambiguity is not desireable[3]. The following derives a condition for the $\mathcal{H}_\mathrm{C}$-representation so that both polytopes necessarily share the same position.

Given is a vector $\mathbf{h}$ that fulfills the constraint $\mathbf{M}_\mathrm{C}\mathbf{h} = \mathbf{0}$. A translation along a vector $\mathbf{x}_0$ to yield a new vector $\mathbf{h}'$ is calculated by:

$$\mathbf{h}' = \mathbf{h} + \mathbf{A}\mathbf{x}_0. \tag{3.5}$$

[3] See the motivation above, but also consider that the algorithm for Minkowski decomposition shall be applied which requires a fixed position of polytopes (see subsection 2.4.2).

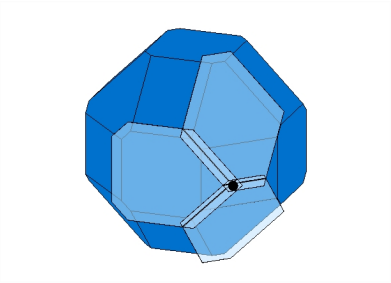

Figure 3.1.: Three supporting hyperplanes (light blue) that intersect in a single point marked by the black spot.

If the translation along $\mathbf{x}_0$ is not prohibited by the $\mathcal{H}_C$-representation, the translated vector $\mathbf{h}'$ also fulfills the constraint which yields:

$$\mathbf{M}_C\mathbf{h}' = (\mathbf{M}_C\mathbf{A})\,\mathbf{x}_0 = \mathbf{0} \tag{3.6}$$

for which $\mathbf{h}'$ is substituted by equation 3.5. It follows that no translation is possible and the position of the represented polytopes is fixed for:

$$\operatorname{rank}(\mathbf{M}_C\mathbf{A}) = 3. \tag{3.7}$$

Fixing the position for $\mathcal{H}_C$-representations. In the rare case that the position is not already fixed[4], the constraints in $\mathbf{M}_C$ can be extended by one or more additional constraints $h_i = 0$. To identify suitable facet indices, valid translation vectors must be found, initially. They can be obtained by a singular value decomposition of $\mathbf{M}_C\mathbf{A}$ and result in one to three orthogonal directions $\mathbf{x}_{0,k}$, dependent on the translational degrees of freedom. Then, for every direction $\mathbf{x}_{0,k}$, a facet with $\langle \mathbf{a}_{i(k)}, \mathbf{x}_{0,k} \rangle \neq 0$ is chosen to be fixed to $h_{i(k)} = 0$ while the normals of the selected facets are linearly independent. The constraint matrix $\mathbf{M}_C$ can be extended with rows $h_{i(k)} = 0$ accordingly and the rows of the selected facets can be set to zero for the group mapping matrix. While the corrected $\mathcal{H}_C$-representation has a reduced dimension n_C, the original set of represented shapes remains covered.

Ill-conditioned representation. Since a representation can be adopted to fulfill equations 3.4 and 3.7 without limiting the range of represented shapes, representations that do not fulfill equation 3.4 or 3.7 are called ill-conditioned. For the remainder of this work, any $\mathcal{H}_C$-representation is assumed to fulfill equations 3.4 and 3.7.

3.2.2 Validity

Motivation. In the introduction to this chapter, an inherent complexity of shape representations was pointed out that is manifested by possible but inconsistent values of the shape coordinates. This issue is revisited in this subsection based on the chosen $\mathcal{H}_C$-representation. The corresponding property of $\mathbf{h}_C$-vectors is defined and the geometrical state space is analyzed accordingly.

[4] Lactose is a sample system that requires position fixing [Dincer, 2000, Choscz, 2012] (see figure 1 in the introduction or figure 3.7 on page 3.7; the facet distances of the $\{010\}$ and $\{0-10\}$ are independent).

Example. Figure 3.2 shows two-dimensional polytopes from the $\mathcal{H}_\mathrm{C}$-representation with the facet normal matrix:

$$\mathbf{A} = \begin{bmatrix} 1 & -1 & 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ 0 & 0 & 1 & -1 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}^T. \tag{3.8}$$

and the mapping matrix:

$$\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}^T. \tag{3.9}$$

The facet distance $h_{\mathrm{C},1}$ belongs to the facets of the square (shape in the middle right) and the facet distance $h_{\mathrm{C},2}$ belongs to the facets of the cross-polytope (shape in the middle left). Starting with a $\mathbf{h}_\mathrm{C}$-vector like for the shape in the middle and increasing $h_{\mathrm{C},1}$ up to the extreme case in the middle left of figure 3.2 results in a polytope where the square facets have grown out. Further increasing $h_{\mathrm{C},1}$ above this limit, like it is shown in the outer left example, does not anymore alter the shape compared to the limiting case. An analogue description could be given for an increasing $h_{\mathrm{C},2}$-coordinate and the right part of figure 3.2. It is assumed now that a simulation of a particle evolution has obtained a $\mathbf{h}_\mathrm{C}$-vector like in the outer left example in figure 3.2 while the growth rate is changed so that $h_{\mathrm{C},1}$ is constant and $h_{\mathrm{C},2}$ increases in size. In such a situation, a real crystal would instantly exhibit the facets of the square again. In the simulation, this reformation of the square facets would be delayed. For that reason, situations of $\mathbf{h}_\mathrm{C}$-vectors like in the outer examples of figure 3.2 are undesirable and the corresponding $\mathbf{h}_\mathrm{C}$-vectors will be called invalid. This validity can be identified by the consistency between the facet distances h_i and the corresponding support measures $h\left(P(\mathbf{h}_\mathrm{C}), \mathbf{a}_i\right)$. If the facet distances and support measures are equal for all facets, then the corresponding facets would also reappear instantly by a change in the growth conditions. If this consistency is violated for a specific facet, a delay would be observed for the reformation of that facet.

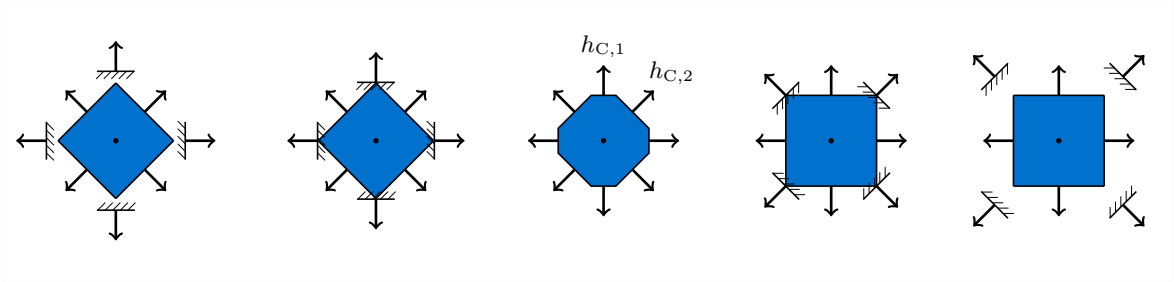

Figure 3.2.: Visualization of the $\mathcal{H}_\mathrm{C}$-representation with the facet normals from equation 3.8 and the group mapping matrix from equation 3.9.

Validity. The validity property illustrated by the example above is first defined for h-vectors. Given a $\mathcal{H}$-representation, a vector $\mathbf{h}$ is called *valid* if each facet distance in $\mathbf{h}$ equals the support function according to the corresponding facet normal:

$$h_i = h(P(\mathbf{h}),\, \mathbf{a}_i). \tag{3.10}$$

This corresponds to the middle, middle left and middle right cases in figure 3.2. Every other vector will be called *invalid* (outer cases in figure 3.2). Since $\mathbf{h}$-vectors might represent empty polytopes for which the support function is not defined, such $\mathbf{h}$-vectors are explicitly defined to be invalid.

Equation 3.10 cannot be given in terms of a $\mathbf{h}_\mathrm{C}$-vector since a component $h_{\mathrm{C},i}$ not necessarily represents a facet distance h_i for general mapping matrices $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}$. However, validity applies likewise to $\mathbf{h}_\mathrm{C}$-vectors via: $\mathbf{h} = \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}\mathbf{h}_\mathrm{C}$ and a $\mathbf{h}_\mathrm{C}$-vector is valid if and only if the corresponding $\mathbf{h}$-vector is valid. Additionally, the term validity can also be applied for facets. If facet i is invalid for a given $\mathbf{h}$- or $\mathbf{h}_\mathrm{C}$-vector, then equation 3.10 is not fulfilled for this specific facet.

Towards the geometrical state space. While equation 3.10 is suitable to determine the validity of a single $\mathbf{h}$- or $\mathbf{h}_\mathrm{C}$-vector, the effort to compute the required support measures is relatively high for large sets of vectors. Hence, we now focus on a method to identify the region of valid $\mathbf{h}$- or $\mathbf{h}_\mathrm{C}$-vectors directly. This problem is already solved for $\mathcal{H}$-representations in mathematical literature and entitled by *'validity of support hyperplane data'* or *'convex set reconstruction'* [Prince and Willsky, 1990, 1991, Karl et al., 1995]. The corresponding algorithm has been explained in subsection 2.4.3. The following only transfers this result to the $\mathbf{h}_\mathrm{C}$-space.

Validity cone. The facet normal matrix $\mathbf{A}'_\mathrm{V}$ from the algorithm to calculate the region of valid $\mathbf{h}$-vectors is casted to the space of $\mathbf{h}_\mathrm{C}$-vectors:

$$\mathbf{A}_\mathrm{V} = \mathbf{A}'_\mathrm{V}\,\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}. \tag{3.11}$$

The region of valid $\mathbf{h}_\mathrm{C}$-vectors is then provided by:

$$\mathbf{A}_\mathrm{V}\mathbf{h}_\mathrm{C} \leq \mathbf{0} \tag{3.12}$$

which is a $\mathcal{H}$-representation of a convex cone in $\mathbf{h}_\mathrm{C}$-space[5]. The cone defined by equation 3.12 will be called cone of valid $\mathbf{h}_\mathrm{C}$-vectors or simply validity cone and it is denoted by $\mathcal{C}_\mathrm{V}$ (see figure 3.3 which is discussed below).

Facet validity cones. Since the underlying algorithm is based on the appearance and disappearance of facets (see subsection 2.4.3), the regions in $\mathbf{h}_\mathrm{C}$-space for which a specific facet is present can also be provided. The transfer of these regions to the $\mathbf{h}_\mathrm{C}$-space is analogous to equations 3.11 and 3.12. The resulting cones are called facet validity cones and they are denoted by $\mathcal{C}_{\mathrm{FV}(i)}$ with facet normal matrices $\mathbf{A}_{\mathrm{FV}(i)}$. For indices $i \in \{1,\ldots,n_\mathrm{H}\}$, the facet i is present at the crystal surface for $\mathbf{h}_\mathrm{C}$ being in the interior of the cone $\mathcal{C}_{\mathrm{FV}(i)}$. The facet has just disappeared for $\mathbf{h}_\mathrm{C}$ being at the boundary of the cone $\mathcal{C}_{\mathrm{FV}(i)}$. And, finally, the facet causes invalidity of the $\mathbf{h}_\mathrm{C}$-vector when it is outside of the cone $\mathcal{C}_{\mathrm{FV}(i)}$. This also implies that the validity cone is the intersection of all facet validity cones:

$$\mathcal{C}_\mathrm{V} = \bigcap_i \mathcal{C}_{\mathrm{FV}(i)}. \tag{3.13}$$

[5]Note that, prior to applications of equation 3.12, redundant inequalities should be eliminated by removing corresponding rows in the matrix $\mathbf{A}_\mathrm{V}$. For example, a lot of rows in $\mathbf{A}_\mathrm{V}$ are identical by symmetry.

Zero-size boundary. The cone $\mathcal{C}_{\mathrm{FV}(0)}$ indicates, similar to subsection 2.4.3, the region of $\mathbf{h}_\mathrm{C}$-vectors that represent non-empty polytopes. A polytope vanishes for $\mathbf{h}_\mathrm{C} \notin \mathcal{C}_{\mathrm{FV}(0)}$ so that the boundary of the non-empty polytope cone also identifies the zero-size boundary at which crystals disappear.

Example validity region. The validity cone for the setup, used in the beginning of this subsection is illustrated in the left of figure 3.3. The validity cone is drawn in light blue and any white area corresponds to invalid $\mathbf{h}_\mathrm{C}$-vectors. The blue sample shapes show how limiting cases at the boundary of the validity cone are obtained. The horizontally aligned shapes demonstrate the increase or decrease of the $h_{\mathrm{C},1}$-coordinate that is assigned to the facets of the square and the vertically aligned shapes demonstrate the increase or decrease of the $h_{\mathrm{C},2}$-coordinate that is assigned to the facets of the cross-polytope. Note that all facets are present for shapes inside of the validity cone while one set of facets disappears for shapes at the boundary of the validity cone.

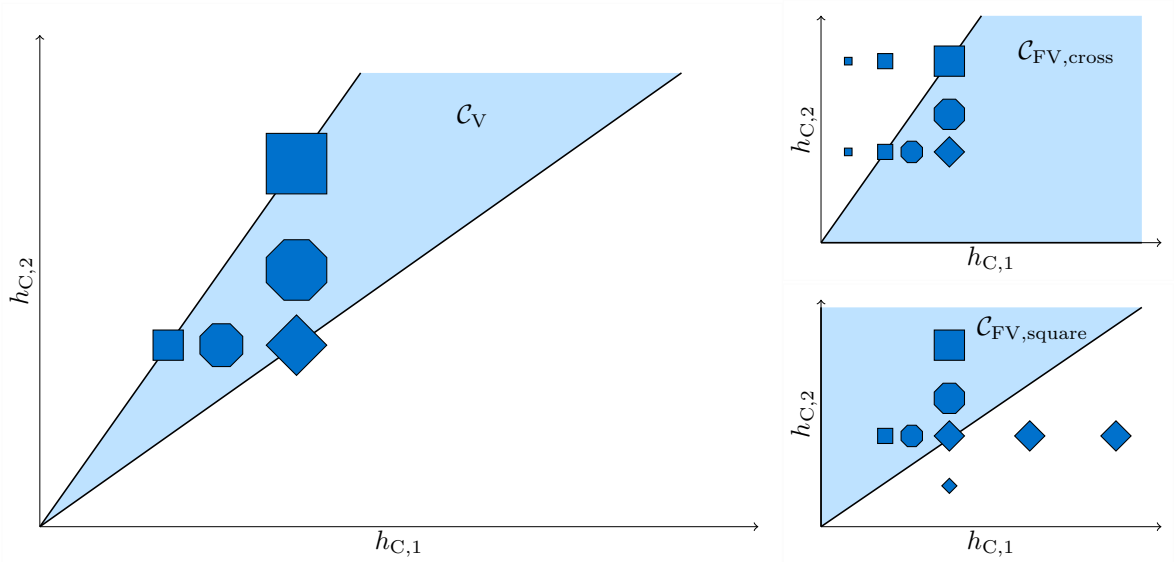

Figure 3.3.: left: validity cone (light blue) and some sample shapes (blue); top right: facet validity cone for any facet of the cross-polytope; bottom right: facet validity cone for any facet of the square

The facet validity cones $\mathcal{C}_{\mathrm{FV}(i)}$ are highlighted in the right of figure 3.3. In the top right, the facet validity cone for any facet from the cross polytope is shown. In the bottom right, the facet validity cone for any facet from the square polytope is shown. The cone of non-empty polytopes is given by the complete positive quadrant of the coordinate system.

Related literature. Distinguishing valid and invalid h-vectors is analogous to the identification of disappearing crystal faces, which was first discussed by Johnsen [1900]. A detailed discussion on the evolution of disappearance conditions for crystal faces is given on page 4 in the introduction. These conditions evolved from a local or simplified analysis to arbitrary crystal shapes [Prywer, 1996, Zhang et al., 2006] and further to the analysis of the complete geometrical state space at once [Borchert, 2007, 2012]. The underlying algorithm could be adopted from the work of Prywer [1996] while, there, the

notation is based on angles. It can also be adopted from the work of Zhang et al. [2006] which was done by Borchert [2007]. His resulting algorithm is analog to the algorithm in this present work but fails when the validity cone comprises a common boundary with the non-empty polytope cone additional to $\mathbf{h}_C = \mathbf{0}$. The reason for this is that the complete disappearance of a crystal is not considered by Zhang et al. or Borchert. For the underlying algorithm in subsection 2.4.3 this consideration is added based on the assumptions on the applicability of that algorithm that were given by Karl et al. [1995]. In a later work by Borchert [2012], he proposed an algorithm for the computation of the validity cone that is completely different. However, also for that algorithm the facet validity cones, the empty-polytope cone are not available and an interpretation of $\mathbf{h}_C$-vectors outside of the validity cone is not possible.

3.2.3 Validity Mapping

Motivation. The analysis of the geometric state space typically assumes valid $\mathbf{h}_C$-vectors since the components of invalid $\mathbf{h}_C$-vectors are not consistent with the support measures of the represented polytope. Therefore, mapping an invalid $\mathbf{h}_C$-vector to a corresponding valid $\mathbf{h}_C$-vector while retaining the shape is an elementary task. Usually, the validity of $\mathbf{h}_C$-vectors is retained in dynamic simulations (see chapter 4). Though, achieving this behavior in dynamic simulations will use the solution of the aforementioned more elementary problem. Additionally, model reduction and shape approximation will only be possible by introducing invalid $\mathbf{h}_C$-vecors which is shown in chapter 6.

Towards validity mapping. The following part derives a procedure to map an arbitrary invalid $\mathbf{h}_C$-vector to a corresponding valid $\mathbf{h}_C$-vector while retaining the shape of the represented polytope. This operation will be denoted by validity mapping. Following equation 3.10, the invalidity of a $\mathbf{h}_C$-vector is caused by the invalidity of certain facets. It is, hence, suitable to correct invalid facets separately to obtain a valid $\mathbf{h}_C$-vector. Correcting a facet while simultaneously retaining the represented polytope shape is, therefore, also called validity mapping.

Coherent facets. Helpful for the following discussion is a new term that addresses the symmetry that also transfers to the validity of facets. It assigns the facets into groups of facets that are either all present or that all have disappeared for a given $\mathbf{h}_C$-vector. It is defined as a binary relation between two facets i and j. Two facets i and j are called coherent, when their facet validity cones are equivalent: $\mathcal{C}_{\mathrm{FV}(i)} = \mathcal{C}_{\mathrm{FV}(j)}$. A facet is always coherent to itself. Additionally, if the validity mapping of a facet is possible (or impossible), it is simultaneously possible (or impossible) for all coherent facets. If the validity mapping is actually applied to a facet, it is simultaneously applied to all coherent facets. For representations that originate from crystal symmetry, facets of the same facet group (or form) are always coherent to each other [Borchardt-Ott, 2009].

Example (coherent facets). Figure 3.4 demonstrates a $\mathcal{H}_C$-representation for which the facet groups according to equal rows in the mapping matrix:

$$\mathbf{M}_{\mathbf{h}_C \mapsto \mathbf{h}} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}^T. \tag{3.14}$$

are not equal to the groups of coherent facets. The facet normal matrix is given by equation 3.8. In figure 3.4, the coordinate $h_{C,1}$ increases from the left to the right while $h_{C,1}$ is kept constant. Beginning with the inner left shape, the first disappearing facets are the top and right facets of the square. Second are the bottom and left facets that just disappear in the outer right sample shape. Never disappearing is the lower left facet of the cross polytope that is also assigned to $h_{C,1}$. The three facets of the cross polytope that are assigned to $h_{C,2}$ could grow out simultaneously as it is evident by the left shape in figure 3.4, assuming a decrease of the coordinate $h_{C,1}$ and going from the right to the left. In total, four groups of coherent facets exist:

1. the three facets assigned to $h_{C,2}$ (disappearing in the left);
2. the top and right facets of the square (disappearing in the inner right);
3. the bottom and left facets of the square (disappearing in the outer right);
4. the lower left facet of the cross-polytope (never disappearing).

In contrast, only two groups of facets exist according to the rows of the group mapping matrix. Additionally, for the invalid $\mathbf{h}_C$-vector that is indicated in the outer right of figure 3.4, no corresponding valid $\mathbf{h}_C$-vector exists that retains the shape. The coherence of facets is, hence, an important property with respect to validity mapping.

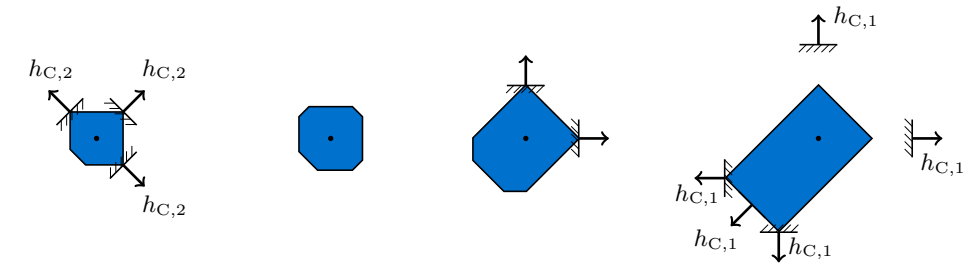

Figure 3.4.: Polytopes according to the facet normal matrix in equation 3.8 and the mapping matrix from equations 3.14

Empty polytopes. An exception for the general validity mapping are invalid $\mathbf{h}_C$-vectors that represent empty polytopes ($P(\mathbf{h}_C) = \emptyset$), similar to the definition of validity itself. Such vectors cannot be mapped to a valid $\mathbf{h}_C$-vector and simultaneously preserve the represented shape since all $\mathbf{h}_C$-vectors with $P(\mathbf{h}_C) = \emptyset$ are invalid by definition. Such vectors are, instead, mapped to $\mathbf{h}_C = \mathbf{0}$ which always represents the polytope $P = \{\mathbf{0}\}$ so that this choice is consistent with any length, area or volume measure. There is no assumption required for this mapping and vectors with $\mathbf{h}_C \notin \mathcal{C}_{FV(0)}$ are excluded in the subsequent derivation.

Example. To support the following derivation, an example is used to outline the basic principles of validity mapping. Figure 3.5 shows an arbitrary validity cone (light blue) with an invalid vector $\mathbf{h}_C$. For this example, it is assumed that each facet of the polytope is represented by either $h_{C,1}$ or $h_{C,2}$. This prohibits a row of the group mapping matrix containing more than one entry that is non-zero (for example: $h_i = h_{C,1} + h_{C,2}$) which is applicable for general $\mathcal{H}_C$-representations.

First, a facet belonging to the group $h_{\mathrm{C},1}$ must have grown out because $h_{\mathrm{C},1}$ is too large in comparison to $h_{\mathrm{C},2}$. Additionally, the facets corresponding to $h_{\mathrm{C},2}$ are assumed still being present and the polytope $P(\mathbf{h}_\mathrm{C})$ is not empty. Secondly, the vector $\mathbf{h}_\mathrm{C}$ must be shifted along a horizontal line (drawn in black) to retain the facet distances corresponding to $h_{\mathrm{C},2}$. Thirdly, the subdomain of the boundary $\partial\mathcal{C}_\mathrm{V}$ to which the invalid vector must be shifted is depicted with a dashed line. Finally, the intersection of the horizontal line, the subdomain where shifting $\mathbf{h}_\mathrm{C}$ is allowed, and the dashed line, the target subdomain of $\partial\mathcal{C}_\mathrm{V}$, determine the proper valid vector $\mathbf{h}'_\mathrm{C}$.

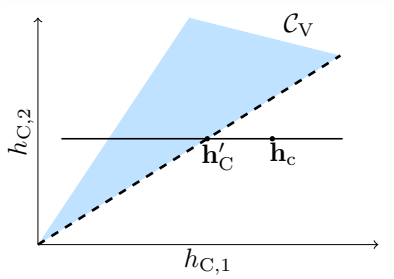

Figure 3.5.: Mapping an invalid vector $\mathbf{h}_\mathrm{C}$ back to the boundary of the validity cone $\partial\mathcal{C}_\mathrm{V}$ to obtain a valid vector $\mathbf{h}'_\mathrm{C}$ with $P(\mathbf{h}) = P(\mathbf{h}')$

Following the above description, the presented viewpoint is entirely based on the $\mathbf{h}_\mathrm{C}$-space. Real polytopes or relations in the corresponding 3-dimensional space are not mentioned. This relatively abstract concept propagates to the derivations below and allows to focus on the relevant property of validity.

Identifying invalid facets. According to the example above, identifying invalid and valid facets is the first step for validity mapping. Note also that, for the definition of validity mapping, it is already suggested that invalid vectors can be corrected facet by facet. The proposed algorithm even goes a step further and resolves each inequality $\mathbf{a}_{\mathrm{FV}(i),j}\mathbf{h}_\mathrm{C} \leq 0$ for the facet validity cones separately. Simultaneously, the algorithm assumes that all other non-coherent facets k are present so that neither the polytope shape nor the differences $h_k - h(P(\mathbf{h}_\mathrm{C}), \mathbf{a}_k)$ can be changed. Identifying an invalid facet i based on a specific violated constraint $\mathbf{a}_{\mathrm{FV}(i),j}\mathbf{h}_\mathrm{C} \leq 0$ is the first step for validity mapping that is outlined in the example above.

Target domain. The shifting direction is constructed in the following such that the shape does not change while $\mathbf{a}_{\mathrm{FV}(i),j}\mathbf{h}_\mathrm{C} \geq 0$ holds. Therefore, the selected constraint also provides the target domain to which the $\mathbf{h}_\mathrm{C}$-vector must be shifted as it is indicated in step three in the example above. It is given by the hyperplane: $\mathbf{a}_{\mathrm{FV}(i),j}\mathbf{h}'_\mathrm{C} = 0$.

Mapping directions. Given a certain set of coherent facets being invalid by the selected constraint, finding a direction in which the $\mathbf{h}_\mathrm{C}$-vector can be shifted without affecting the polytope shape is the next required step. Therefore, two paths are possible.

The most obvious way that was already outlined in the example is to affect the distances of the invalid facets while retaining the distances of all non-coherent facets. This is possible, given that the rank of the group mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}$, setting all rows of coherent facets to zero, is smaller than n_C. A mapping direction $\mathbf{h}_{\mathrm{C},\mapsto} \neq \mathbf{0}$ is then ob-

tained by a singular value decomposition of the altered group mapping matrix[6]. Since the direction $\mathbf{h}_{\mathrm{C},\mapsto}$ must affect some facet distances as long as the $\mathcal{H}_{\mathrm{C}}$-representation is not ill-conditioned and since these distances cannot be any of the non-coherent facets, the direction $\mathbf{h}_{\mathrm{C},\mapsto}$ necessarily affects the distances of the coherent facets. This approach changes the distance h_i while retaining the support measure $h(P(\mathbf{h}_{\mathrm{C}}), \mathbf{a}_i)$ so that it is suitable for validity mapping.

A possible alternative to the correction above is the translation of the polytope. This is reasonable given that the position is not fixed any more when the coherent facets are removed from the representation. This condition is fulfilled when the rank of $(\mathbf{M}_{\mathrm{C}}\mathbf{A})$ is lower than 3 when the rows of coherent facets in $\mathbf{A}$ are set to zero. The direction $\mathbf{x}_0$ for the translation is obtained by the singular value decomposition of the altered matrix $(\mathbf{M}_{\mathrm{C}}\mathbf{A})$. The direction in $\mathbf{h}_{\mathrm{C}}$-space is then given by:

$$\mathbf{h}_{\mathrm{C},\mapsto} = \mathbf{M}^{+}_{\mathbf{h}_{\mathrm{C}}\mapsto\mathbf{h}}\mathbf{A}'\mathbf{x}_0 \tag{3.15}$$

where $\mathbf{A}'$ is the adopted facet normal matrix. This operation typically changes all non-coherent facet distances while retaining the distances of the coherent facets. In contrast to the previous path, this approach changes the support measure $h(P(\mathbf{h}_{\mathrm{C}}, \mathbf{a}_i))$ while retaining h_i.

Note that the two paths directly link to the two types of ill-conditioned representations (see subsection 3.2.1). Based on the disappeared set of coherent facets, an altered $\mathcal{H}_{\mathrm{C}}$-representation can be assumed for which these facets do not exist. This implies to delete the corresponding rows of the matrices $\mathbf{A}$ and $\mathbf{M}_{\mathbf{h}_{\mathrm{C}}\mapsto\mathbf{h}}$. The mentioned link is then provided for the first approach by assuming a group mapping matrix with a rank lower than n_{C} and for the second approach by assuming that the position of the represented polytopes is not fixed anymore.

Final correction. The final step performs the mapping by: $\mathbf{h}'_{\mathrm{C}} = \mathbf{h}_{\mathrm{C}} + \gamma\,\mathbf{h}_{\mathrm{C},\mapsto}$ where γ represents (not necessarily equals) the distance of $\mathbf{h}_{\mathrm{C}}$ to the target domain $\mathbf{a}_{\mathrm{FV}(i),j}\mathbf{h}'_{\mathrm{C}} = 0$ and is given by:

$$\gamma = -\frac{\mathbf{a}^T_{\mathrm{FV}(i),j}\,\mathbf{h}_{\mathrm{C}}}{\mathbf{a}^T_{\mathrm{FV}(i),j}\mathbf{h}_{\mathrm{C},\mapsto}} \tag{3.16}$$

to yield:

$$\mathbf{h}'_{\mathrm{C}} = \mathbf{h}_{\mathrm{C}} - \mathbf{a}^T_{\mathrm{FV}(i),j}\,\mathbf{h}_{\mathrm{C}}\left(\frac{1}{\mathbf{a}^T_{\mathrm{FV}(i),j}\mathbf{h}_{\mathrm{C},\mapsto}}\mathbf{h}_{\mathrm{C},\mapsto}\right). \tag{3.17}$$

This equation can be expressed as a projection:

$$\mathbf{h}'_{\mathrm{C}} = \mathbf{P}_{\mathrm{FV}(i),j}\mathbf{h}_{\mathrm{C}} \tag{3.18}$$

and the matrix:

$$\mathbf{P}_{\mathrm{FV}(i),j} = \left(\mathbf{I}_{n_{\mathrm{C}}} - \frac{\mathbf{h}_{\mathrm{C},\mapsto}\mathbf{a}^T_{\mathrm{FV}(i),j}}{\mathbf{a}^T_{\mathrm{FV}(i),j}\mathbf{h}_{\mathrm{C},\mapsto}}\right) \tag{3.19}$$

is called a validity projection matrix.

[6] This direction is uniquely determined up to a scalar coefficient. Since this detail is not essential for the given derivation, it is not discussed.

Overall algorithm. The complete procedure for the validity mapping of a vector $\mathbf{h}_\mathrm{C}$ can be constructed as given by the following.

1. Check all constraints $\mathbf{a}_{\mathrm{FV}(i),j}^T \mathbf{h}_\mathrm{C} \leq 0$.
2. If the vector is valid, the algorithm terminates.
3. If one constraint indicates an empty polytope, set $\mathbf{h}_\mathrm{C} = \mathbf{0}$ and terminate.
4. Otherwise, select a violated constraint, perform the validity mapping according to equation 3.18 and return to step 1.

This procedure is applicable, but not efficient for an implementation or for further derivations. The constraints of the facet validity cones are checked several times in step (1) and a facet might be corrected multiple times when more than one constraint from the corresponding facet validity cone is violated. These issues can be resolved by restructuring the procedure. First, multiple corrections of the same facet can be avoided. Since all corrections shift the $\mathbf{h}_\mathrm{C}$-vector according to the same direction, only one of the projections from equation 3.18 is actually required. This projection can be identified by demanding that the corrected vector resides in the facet validity cone which results in a condition to the input $\mathbf{h}_\mathrm{C}$-vector:

$$\left(\mathbf{A}_{\mathrm{FV}(i)} \mathbf{P}_{\mathrm{FV}(i),j}\right) \mathbf{h}_\mathrm{C} \leq \mathbf{0}. \tag{3.20}$$

Secondly, each correction only affects the corresponding coherent facets so that the validity of non-coherent facets is not affected. Given that each facet can now be corrected in a single step, the constraints must only be evaluated once. The validity mapping can now be restructured to the following algorithm.

1. Check all constraints $\mathbf{a}_{\mathrm{FV}(i),j}^T \mathbf{h}_\mathrm{C} \leq 0$.
2. If one constraint indicates an empty polytope, set $\mathbf{h}_\mathrm{C} = \mathbf{0}$ and terminate.
3. For every invalid group of coherent facets, identify the appropriate validity projection matrix according to equation 3.20 and apply it according to equation 3.18.

The efficiency is further increased by checking only unique constraints for step (1) and computing the projection matrices $\mathbf{P}_{\mathrm{FV}(i),j}$ and constraint matrices $\mathbf{A}_{\mathrm{FV}(i)} \mathbf{P}_{\mathrm{FV}(i),j}$ *a priori*.

Finally, the requirements for the applicability of the algorithm are summarized. Validity mapping of a vector $\mathbf{h}_\mathrm{C}$ is possible, when

1. the represented polytope is empty ($\mathbf{h}_\mathrm{C} \notin \mathcal{C}_{\mathrm{FV}(0)}$) or
2. when for every invalid facet i either:
   a) the rank of the matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C} \mapsto \mathbf{h}}$ for which all rows of coherent facets are set to zero is lower than n_C or
   b) when the rank of the matrix $\mathbf{M}_\mathrm{C} \mathbf{A}$, given that the rows of coherent facets in $\mathbf{A}$ are set to zero, is lower than 3.

In any other case, the polytope is non-empty (condition 1) while no direction in $\mathbf{h}_\mathrm{C}$-space exists that does not alter the shape of the polytope (condition 2). Validity mapping is not possible for such cases[7]. One example of a $\mathbf{h}_\mathrm{C}$-vector for which validity mapping is not possible is given by the outer right shape in figure 3.8[8].

Further simplifications. The discussion above suggests that the mapping of an invalid $\mathbf{h}_\mathrm{C}$-vector is possible with a single projection. Therefore, it would be required to create all combinations of projection matrices $\mathbf{P}_{\mathrm{FV}(i),j}$ including the accumulation of the constraints for their applicability. The total number of such combinations increases rapidly with n_C so that calculating these projection matrices *a priori* is not reasonable. However, this concept is still valuable for derivations so that the overall validity projection matrix is denoted by $\mathbf{P}_\mathrm{V}(\mathbf{h}_\mathrm{C})$ and the corresponding required conditions by $\mathbf{A}_{\mathbf{P}_\mathrm{V}}(\mathbf{h}_\mathrm{C})$. These conditions include the constraints from equation 3.20 and the specifically violated constraint of the facet validity cone: $\mathbf{a}_{\mathrm{FV}(i),j}^T\mathbf{h}_\mathrm{C} \leq 0$.

3.2.4 Proper Representations

Motivation. Since validity mapping is not always possible, it follows immediately that two classes of $\mathcal{H}_\mathrm{C}$-representations can be separated. One for which validity mapping is possible for all $\mathbf{h}_\mathrm{C}$-vectors and another one for which this is not the case. This subsection defines this classification and provides corresponding conditions that are applicable computationally. Therefore, the viewpoint changes from specific $\mathbf{h}_\mathrm{C}$-vecors (or polytopes) to $\mathcal{H}_\mathrm{C}$-representations (or classes of polytopes), compared to the previous subsection.

Proper $\mathcal{H}_\mathrm{C}$-representations. $\mathcal{H}_\mathrm{C}$-representations for which validity mapping is possible for arbitrary $\mathbf{h}_\mathrm{C}$-vectors are called *proper*.[9] Contrarily, when a $\mathbf{h}_\mathrm{C}$-vector exists for which validity mapping cannot be applied, this representation is called *improper*.[10] Ill-conditioned representations are neither proper nor improper, they are a third class of $\mathcal{H}_\mathrm{C}$-representations that are not treated for the reasons given in subsection 3.2.1.

Properness conditions. To decide the properness of a given $\mathcal{H}_\mathrm{C}$-representation computationally, the conditions for a successful validity mapping from the previous subsection are reformulated. A $\mathcal{H}_\mathrm{C}$-representation is proper and guarantees a successful validity mapping when for each group of coherent facets, either

1. the facet validity cones equal the non-empty polytope cone $\mathcal{C}_{\mathrm{FV}(i)} = \mathcal{C}_{\mathrm{FV}(0)}$, or
2. the rank of the matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}$ for which all rows of coherent facets are set to zero is lower than n_C, or
3. the rank of the matrix $\mathbf{M}_\mathrm{C}\mathbf{A}$, given that the rows of coherent facets in $\mathbf{A}$ are set to zero, is lower than 3.

Condition (1) corresponds to the validity mapping of empty polytopes $P(\mathbf{h}_\mathrm{C}) = \emptyset$, condition (2) links to the mapping that affects the facet distances of the invalid facet i while condition (3) enables the mapping that is based on a translation of the polytope.

[7] Though, for $\mathbf{h}_\mathrm{C}$-vectors that originate from proper crystal symmetry, validity mapping is always possible as it will be discussed in the subsequent subsection.

[8] Another example is introduced in subsection A.2.3 since cases not fulfilling the above constraints require $\mathcal{H}_\mathrm{C}$-representations apart from the typical crystal symmetry.

[9] An example of a proper representation is given by figure 3.2.

[10] An example of an improper representation is given by figure 3.4.

Crystal representations. Whenever a $\mathcal{H}_{\mathrm{C}}$-representation originates from crystal symmetry, it always fulfills the conditions for properness. The symmetry of crystals guarantees that each face of a crystal form is either present or has disappeared so that faces of a crystal form are coherent [Borchardt-Ott, 2009]. When the $\mathcal{H}_{\mathrm{C}}$-representation is formulated, each crystal form corresponds to a component of the $\mathbf{h}_{\mathrm{C}}$-vector and each row of the group mapping matrix contains a single one. If a crystal form grows out and the rows of invalid coherent facets are set to zero in the group mapping matrix, the rank is always $n_{\mathrm{C}} - 1$. This is already sufficient to conclude that an $\mathcal{H}_{\mathrm{C}}$-representation that originates from crystal symmetry are always proper, even if the position of the corresponding polytopes might not be fixed. Fixing the position by setting one or more face distances to 0, as described in the subsection 3.2.1, immediately implies that these faces can be fixed by moving the whole polytope as described in the preceding subsection.

Relevance of proper $\mathcal{H}_{\mathrm{C}}$-representations. All applications in this work concern crystallization processes so that proper $\mathcal{H}_{\mathrm{C}}$-representations constitute the native focus. Additionally, even in the cases where improper $\mathcal{H}_{\mathrm{C}}$-representations are discussed, corresponding proper $\mathcal{H}_{\mathrm{C}}$-representations are exploited to utilize or analyze them (see chapter 6). Therefore, improper representations are only analyzed in appendix A.2 and the following sections assume proper $\mathcal{H}_{\mathrm{C}}$-representations if not noted, otherwise.

3.3 Calculating Measures

3.3.1 Introduction

Motivation. Subsection 3.1.1 identifies three general problems for crystal shape modeling. The first two issues, the validity of vectors in the crystal state space and disappearing polytopes, are resolved in section 3.2. The third issue, discussed in this section, concerns measure calculations. It was pointed out in subsection 2.3.1 that measure calculation strongly depends on the present morphology, or a-type. Corresponding regions in the $\mathbf{h}_\mathrm{C}$-space with equal morphology are, however, not yet identified. At this point, the incorporation of the convex geometry to the analysis and derivations becomes profitable. The Minkowski decomposition (see subsections 2.4.1) and the corresponding $\mathcal{S}$-representation (see subsection 2.2.4) are appropriate tools to handle morphology based problems. Additionally, the concept of mixed volumes (see subsection 2.3.2) links the $\mathcal{S}$-representation with measure computation and, finally, the literature provides that a variety of measures can be treated analogously (see subsection 2.3.3).

Outline. In the presented approach, it is aimed to transfer the measure calculation by mixed volumes for polytopes in $\mathcal{S}$-representation to equations that are applied directly to polytopes given in $\mathcal{H}_\mathrm{C}$-representation. Therefore, the Minkowski decomposition that is given for $\mathcal{H}$-representations, is adopted for $\mathcal{H}_\mathrm{C}$-representations to solve an auxiliary problem in subsection 3.3.2. This auxiliary problem discloses how a-types are represented in $\mathbf{h}_\mathrm{C}$-space. Subsequently, subsection 3.3.3 introduces a decomposition of the validity cone based on this auxiliary problem. Given this decomposition, all structuring elements are identified that are required to represent all polytopes of a $\mathcal{H}_\mathrm{C}$-representation in $\mathcal{S}$-representation. This decomposition is also essential for any problem that is related to the surface structure of the represented polytopes since it identifies all present a-types. Subsection 3.3.4 then provides a coordinate transformation between the $\mathcal{S}$- and $\mathcal{H}_\mathrm{C}$-representation. This, finally, allows to convey the concepts of convex geometry to the $\mathbf{h}_\mathrm{C}$-space, including in particular the measure computation by mixed volumes. The newly obtained framework for measure calculation is described and summarized in subsection 3.3.5.

Alternative. An alternative to the approach above would be to directly deduce polynomials in the components of the $\mathbf{h}_\mathrm{C}$-vectors without introducing Minkowski addition or mixed volumes. An example is shown for the polytope volume in subsection 2.3.1. However, the proposed approach to start with mixed volumes has benefits, while none are available for the alternative. The concept of mixed volumes is readily available for several measures without additional derivations (see subsection 2.3.2) and the required coefficients are only dependent on 1, 2 or 3 structuring elements. Overall, it can be presumed that calculating the mixed volumes by evaluating the measures for generated sample polytopes is less effort and more general than deriving the polynomials in components of the $\mathbf{h}_\mathrm{C}$-vectors directly. Additionally, the coefficients for the polynomials in components of the $\mathbf{h}_\mathrm{C}$-vector are only valid for a specific a-type. Following subsection 2.4.2, finding the region in the crystal state space that is covered by a certain a-type is essentially the same problem as identifying the structuring elements to express all polytopes of that a-type in $\mathcal{S}$-representation. Hence, this alternative cannot reduce the complexity to obtain an efficient framework for measure calculation. In fact, the derivations in this and the following subsection result in a polynomial of the $\mathbf{h}_\mathrm{C}$-vector components

and a linear inequality that expresses the region for certain a-types. However, while a result that matches the problem statement of the alternative approach is obtained, this result is achieved indirectly via the introduction of $\mathcal{S}$-representations.

3.3.2 Constrained Minkowski Decomposition

Motivation. This subsection introduces and solves an auxiliary problem that is required to link $\mathcal{H}_C$-representations with $\mathcal{S}$-representations. This allows, in following subsections, to utilize the findings from convex geometry which includes, in particular, the concept of mixed volumes. The statement of the auxiliary problem is: given a polytope Q in $\mathcal{H}_C$-representation, find a set of structuring elements P_i so that all polytopes $P(\mathbf{h}_C) \leq Q$ that fulfill the constraints of the $\mathcal{H}_C$-representation are also covered by this $\mathcal{S}$-representation.

Prior results. A problem analogue to the auxiliary problem in this subsection is already solved in subsection 2.4.2 to link the $\mathcal{S}$-representation with the $\mathcal{H}$-representation. Given a polytope Q in $\mathcal{H}$-representation, all polytopes $P \leq Q$ that are summands of Q can be expressed in an $\mathcal{S}$-representation that uses the indecomposable polytopes $P_i \leq Q$ as structuring elements. An algorithm, denoted by Minkowski decomposition, is provided in subsection 2.4.2 to calculate these indecomposable polytopes. The main result of this algorithm is a cone in $\mathbf{h}$-space that represents all polytopes $P \leq Q$. Given that cone, the indecomposable polytopes are represented by the extreme rays of this cone.

Interpretation of the auxiliary problem. The auxiliary problem contains an additional constraint compared to the problem solved by Minkowski decomposition: the summands $P \leq Q$ must be representable in $\mathcal{H}_C$-representation. This implies that any $\mathbf{h}$-vector must fulfill the constraint $\mathbf{M}_C\mathbf{h} = \mathbf{0}$ with $\mathbf{M}_C$ being the constraint matrix of the $\mathcal{H}_C$-representation. The set of $\mathbf{h}$-vectors that fulfill this constraint constitute a linear subspace of the $\mathbf{h}$-space, the null-space of the matrix $\mathbf{M}_C$. Alternatively, the same subspace is given by the mapping matrix $\mathbf{M}_{\mathbf{h}_C \mapsto \mathbf{h}}$: the columns of the matrix $\mathbf{M}_{\mathbf{h}_C \mapsto \mathbf{h}}$ are a basis of this subspace.

Solution of the auxiliary problem. It follows that the set of polytopes that must be covered by the $\mathcal{S}$-representation of the auxiliary problem are represented in $\mathbf{h}$-space by the intersection of the cone from Minkowski decomposition and the null space of the constraint matrix $\mathbf{M}_C$. This intersection is necessarily a convex cone and any vector $\mathbf{h}_i$ of the cone represents a summand $P(\mathbf{h}) \leq Q$ so that $\sum_i \lambda_i P(\mathbf{h}_i) = P(\sum_i \lambda_i \mathbf{h}_i)$ holds for $\lambda_i \geq 0$. The extreme rays of that cone consequently represent the required structuring elements for the auxiliary problem. This conclusion is analogue to the discussion of Minkowski decomposition in subsections 2.4.1 and 2.4.2.

While the structuring elements of the auxiliary problem are not necessarily indecomposable they are still special in the sense that, for each structuring element, the set of existing edges cannot be reduced any further.

Constrained Minkowski decomposition. As the Minkowski decomposition already uses linear constraints, the additional constraint $\mathbf{M}_C\mathbf{h} = \mathbf{0}$ can be concatenated with these existing ones (see subsection 2.4.2). Calculating the solution of the auxiliary problem is then analogue to Minkowski decomposition. This alteration of the original algorithm will be called constrained Minkowski decomposition and the primary result is the convex cone for which (exactly) all $\mathbf{h}$-vectors of that cone are summands $P \leq Q$

and can be rewritten in $\mathcal{H}_\mathrm{C}$-representation.

Fixed positions. A $\mathcal{H}_\mathrm{C}$-representation guaranties normalized polytope positions (see subsection 3.2.1) so that the additional constraints that were necessary for the Minkowski decomposition are not required for constrained Minkowski decomposition. The constraints are already contained in the matrix $\mathbf{M}_\mathrm{C}$.

The cone in $\mathbf{h}_\mathrm{C}$-space. The cone that is obtained by constrained Minkowski decomposition is necessarily embedded in the linear subspace defined by the $\mathcal{H}_\mathrm{C}$-representation. Consequently, it is evaluated directly in $\mathbf{h}_\mathrm{C}$-space. Likewise, the input polytope Q is directly given with a $\mathbf{h}_\mathrm{C}$-vector. The resulting cone is usually n_C-dimensional. If a cone is lower dimensional for an input polytope Q', the algorithm for constrained Minkowski decomposition generated an equality constraint for facets that intersect in the same extreme point of Q'. Small perturbations of the $\mathbf{h}_\mathrm{C}$-vector for Q' are sufficient to remove this constraint so that a higher dimensional cone is obtained. In general, choosing the input polytope Q by a random $\mathbf{h}_\mathrm{C}$-vector makes it unlikely to obtain a cone that is not n_C-dimensional.

Maximum a-types. The following considers an n_C-dimensional cone, generated by the polytope Q. The interior of this cone represents all $\mathbf{h}_\mathrm{C}$-vectors for which the polytopes $P(\mathbf{h}_\mathrm{C})$ have the same a-type $[Q]$. At the boundary of this cone, a-types $[P] \leq [Q]$ are found that have a smaller number of edges so that the a-type $[Q]$ will be called a maximum a-type for the given $\mathcal{H}_\mathrm{C}$-representation[11].

3.3.3 Decomposition of the Validity Cone

Motivation. The previous subsection introduced an auxiliary problem to identify the structuring elements for a single input polytope. This subsection presents an algorithm to identify the structuring elements for all polytopes of a $\mathcal{H}_\mathrm{C}$-representation so that the corresponding $\mathcal{S}$-representation covers at least all shapes of the $\mathcal{H}_\mathrm{C}$-representation.

Partitioning. The total number of a-types is finite for combinatorial reasons so that only a finite number of maximum a-types exist. The corresponding n_C-dimensional cones from constrained Minkowski decomposition can, hence, be enumerated and will be denoted by $\mathcal{C}_{\mathrm{U},p}$.

Because the interior of each cone equals all $\mathbf{h}_\mathrm{C}$-vectors with the same a-type, the corresponding cones cannot overlap. Formally, this is expressed with the dimension of the intersection of two cones $p \neq q$:

$$\dim\left(\mathcal{C}_{\mathrm{U},p} \cap \mathcal{C}_{\mathrm{U},q}\right) < n_\mathrm{C}. \tag{3.21}$$

Additionally, given the validity cone which covers all achievable polytopes and a-types of a $\mathcal{H}_\mathrm{C}$-representation, this cone can be decomposed into the cones from constrained Minkowski decomposition that are generated by the finite number of maximum a-types. Given all cones $\mathcal{C}_{\mathrm{U},p}$, it holds:

$$\mathcal{C}_\mathrm{V} = \bigcup_i \mathcal{C}_{\mathrm{U},i}. \tag{3.22}$$

Each cone $\mathcal{C}_{\mathrm{U},i}$ unifies a set of a-types: the maximum a-type $[Q]$ and any a-type $[P] \leq [Q]$ for which the polytopes $P \in [P]$ can be written in $\mathcal{H}_\mathrm{C}$-representation. Therefore, the

[11] In contrast to the term 'a-type' which is frequently used in literature, the term 'maximum a-type' is not used in literature and only valid in the scope of this work.

cones $\mathcal{C}_{\mathrm{U},p}$ are called unified partition and the index U is chosen. Independent of $\mathcal{C}_{\mathrm{U},p}$ being a unified partition obtained from constrained Minkowski decomposition, any set of n_{C}-dimensional cones that fulfill equations 3.21 and 3.22 are called a partitioning of the validity cone.

Example. Figure 3.6 demonstrates a partitioning of the validity cone into two unified partitions $\mathcal{C}_{\mathrm{U},p}$. The chosen example models the facets of a cube (blue in the upper left) for which the facet distances are assigned to $h_{\mathrm{C},1}$. The facet distances that are assigned to $h_{\mathrm{C},2}$ belong to an octahedron (dark blue in the right). Both unified partitions are separated by a third polytope (or structuring element) that can be obtained by a constrained Minkowski decomposition from one of the two small polytopes near the center. These small polytopes also illustrate the maximum a-types that are assigned to the unified partitions (see also subsection 2.3.1).

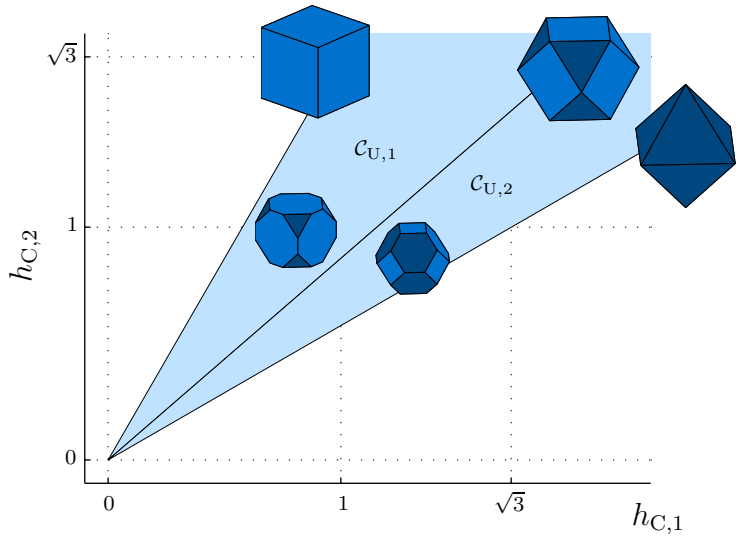

Figure 3.6.: Two unified partitions $\mathcal{C}_{\mathrm{U},p}$ that are separated by a structuring element.

Setup of the decomposition algorithm. Before an algorithm is provided to find all unified partitions for a given $\mathcal{H}_{\mathrm{C}}$-representation, some notation is introduced and basic considerations are discussed.

Storage of results. The unified partitions resulting from the algorithm are stored with their facet normal matrices $\mathbf{A}_{\mathrm{U},p}$. The total number of unified partitions is denoted by n_{PU}. Structuring elements are given with the extreme rays of the unified partitions and are obtained by the double-description method (see appendix A.1.2). All these structuring elements are necessary and sufficient to cover all polytopes of the $\mathcal{H}_{\mathrm{C}}$-representation according to the discussion of the auxiliary problem in the preceding subsection. They are stored in two matrices of $\mathbf{h}_{\mathrm{C}}$-vectors because two different scalings of the structuring elements are required. The total number of structuring elements is denoted by n_{S}.

Scaling of structuring elements. In the following, $\mathbf{h}'_{\mathrm{C},S_k}$ denotes an arbitrary extreme ray element to define the two scalings of the structuring elements.

The first version of the $\mathbf{h}_\mathrm{C}$-vector scales the polytope $P(\mathbf{h}'_{\mathrm{C},S_k})$ to a mean width of 1. This vector is used whenever measures (*e.g.* volume or surface area) are computed so that it is called physically scaled. The physically scaled vector is defined with:

$$\tilde{\mathbf{h}}_{\mathrm{C},S_k} = \frac{1}{\mu^{\text{mean width}}\left(P\left(\mathbf{h}'_{\mathrm{C},S_k}\right)\right)}\mathbf{h}'_{\mathrm{C},S_k}. \tag{3.23}$$

Scaling by other measures like volume or surface area is not reasonable because a structuring element can be just a line segment for some cases. The polytope volume or surface area would be 0 in these cases while the mean width cannot reduce to 0 unless the polytope contains only a single point. This scaling prevents numerical problems (very small or very large polytopes) that could occur for the following alternative scaling.

The second version of $\mathbf{h}_\mathrm{C}$-vectors is scaled for properties in the $\mathbf{h}_\mathrm{C}$-space and will therefore be called natively scaled. The main target for this scaling is that all such scaled $\mathbf{h}_\mathrm{C}$-vectors reside on the same hyperplane. Benefits of this scaling are outlined in the following paragraph. The normal vector of the hyperplane is given with the mean of all physically scaled extreme rays of the validity cone. Scaled to a length of 1, this normal vector is denoted by $\mathbf{h}_{\mathrm{C,center}}$. The natively scaled vectors are then defined with:

$$\bar{\mathbf{h}}_{\mathrm{C},S_k} = \frac{1}{\left\langle \mathbf{h}_{\mathrm{C,center}},\, \mathbf{h}'_{\mathrm{C},S_k} \right\rangle}\mathbf{h}'_{\mathrm{C},S_k}. \tag{3.24}$$

which implies that the distance of that hyperplane from the origin is set to 1. Natively scaled $\mathbf{h}_\mathrm{C}$-vectors have a minimum length of 1.

Closed cones. Natively scaled extreme rays are used to represent convex cones by corresponding polytopes. The polytopes are specified in $\mathcal{V}$-representation with the natively scaled extreme ray elements and the origin as vertices. The $\mathcal{H}$-representation of these polytopes is given by the facet normal matrix of the cone with facet distances of zero while the normal vector $\mathbf{h}_{\mathrm{C,center}}$ with a facet distance of 1 is appended. A polytope that is constructed in that way is called a closed cone and this closure is denoted by a bar: $\bar{\mathcal{C}}$. Converting between the closure $\bar{\mathcal{C}}$ and the cone $\mathcal{C}$ is always possible by adding or removing the closing facet (normal vector $\mathbf{h}_{\mathrm{C,center}}$).

The volume of a closed cone is, for example, used to measure the size of a cone. The main reason to introduce closed cones, however, is an algorithm for set difference which is only implemented for polytopes.[12] The set difference between a domain $\mathcal{D}$ and a unified partition is a central step in the decomposition algorithm.

Decomposition algorithm. The input for the decomposition algorithm is a $\mathcal{H}_\mathrm{C}$-representation: a matrix of facet normals $\mathbf{A}$ and a group-mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\to\mathbf{h}}$ or constraint matrix $\mathbf{M}_\mathrm{C}$. The frame of the decomposition algorithm is an iterative loop that reduces a domain $\mathcal{D}$ in each step by subtracting a new unified partition until this domain is empty. This domain is initialized with the validity cone: $\mathcal{D} = \mathcal{C}_\mathrm{V}$. As the result of a set difference is not necessarily convex, the domain $\mathcal{D}$ is given by a partitioning into convex cones: $\mathcal{D} = \cup_i \mathcal{D}_i$. For the initial state, only one partition $\mathcal{D}_1 = \mathcal{D} = \mathcal{C}_\mathrm{V}$ exists.

[12] The algorithm for set difference is documented in the code. The minuend is given as a set of polytopes in $\mathcal{V}$-representation constituting a triangulation of the minuend. The subtrahend is given or converted to $\mathcal{H}$-representation. The core of the algorithm is then based on a case-by-case analysis how a hyperplane of the subtrahend cuts a simplex of the minuend. The output is the triangulated set difference.

After the above initialization, the algorithm performs the following steps.

1. **Generate random vector** $\mathbf{h}_\mathrm{C} \in \mathcal{D}_1$
   The natively scaled extreme rays of $\mathcal{D}_1$ are multiplied by random numbers and added to obtain the vector $\mathbf{h}_\mathrm{C}$. The random numbers are chosen to be significantly larger than 0 but smaller than 1 to guarantee that $\mathbf{h}_\mathrm{C}$ is not at the boundary of $\mathcal{D}_1$. If $\mathbf{h}_\mathrm{C}$ would be at the boundary of $\mathcal{D}_1$, it is likely that it is also on the boundary of some unified partition.

2. **Constrained Minkowski decomposition**
   A constrained Minkowski decomposition for the polytope $P(\mathbf{h}_\mathrm{C})$ is performed and a corresponding cone is obtained. If the obtained cone is not n_C-dimensional, we return to step 1. This event is unlikely for random vectors $\mathbf{h}_\mathrm{C}$ so that no infinite loop is created (see subsection 3.3.2 for details). If it is n_C-dimensional, a proper partition $\mathcal{C}_{\mathrm{U},p}$ is found.

3. **Accept new partition**
   The structuring elements are stored physically and natively scaled. The partition is also stored with $\mathbf{A}_{\mathrm{U},p}$. The remaining domain is updated by the set difference $\mathcal{D}\backslash\mathcal{C}_{\mathrm{U},p}$ and stored via its partitioning into convex cones: $\mathcal{D} = \cup_i \mathcal{D}_i$.

4. **Termination check**
   If the domain $\mathcal{D}$ is empty, the decomposition is complete according to equation 3.22 and the algorithm terminates. If the domain $\mathcal{D}$ is not empty, it is returned to step 1.

Conclusion. With the decomposition of the validity cone $\mathcal{C}_\mathrm{V}$ into unified partitions $\mathcal{C}_{\mathrm{U},p}$, all structuring elements S_i are determined that are required to describe any polytope $P(\mathbf{h}_\mathrm{C})$ by means of a linear combination of structuring elements $\sum_i \lambda_i S_i$. These structuring elements are the polytopes $P(\tilde{\mathbf{h}}_{\mathrm{C},S_i})$ from the physically scaled extreme ray elements of the unified partition.

Range of the $\mathcal{S}$-representation. Given two arbitrary but distinct unified partitions $\mathcal{C}_{\mathrm{U},p}$ and $\mathcal{C}_{\mathrm{U},q}$, an important insight can be revealed. Given these unified partitions, there is a pair of structuring elements S_k with $\mathbf{h}_{\mathrm{C},S_k} \in \mathcal{C}_{\mathrm{U},p}$ and S_l with $\mathbf{h}_{\mathrm{C},S_l} \in \mathcal{C}_{\mathrm{U},q}$ so that the polytope $S_l + S_k$ is not representable in $\mathcal{H}_\mathrm{C}$-representation. This implies that the $\mathcal{S}$-representation covers more polytopes than the input $\mathcal{H}_\mathrm{C}$-representation when more than one unified partition exists.

The existence of the polytopes S_k and S_l can be shown by contradiction. If this pair does not exist, all structuring elements of both unified partitions can be added to obtain a polytope Q in $\mathcal{H}_\mathrm{C}$-representation. For this polytope, a constrained Minkowski decomposition can be performed and a unified partition $\mathcal{C}_{\mathrm{U},r}$ is obtained. This cone is n_C-dimensional and necessarily covers the unified partitions $\mathcal{C}_{\mathrm{U},p}$ and $\mathcal{C}_{\mathrm{U},q}$ from which the summands are taken to create Q. As these unified partitions also follow a constrained Minkowski decomposition with a resulting n_C-dimensional cone, all partitions $\mathcal{C}_{\mathrm{U},p} = \mathcal{C}_{\mathrm{U},q} = \mathcal{C}_{\mathrm{U},r}$ are equal. This contradicts the assumption that $\mathcal{C}_{\mathrm{U},p}$ and $\mathcal{C}_{\mathrm{U},q}$ are distinct unified paritions so that S_k and S_l must exist.

The constraint $\mathbf{M}_\mathrm{C}\mathbf{h} = \mathbf{0}$ cannot be responsible for $S_k + S_l$ not being representable in $\mathcal{H}_\mathrm{C}$-representation. All structuring elements (given by $\mathbf{h}_{S_i}$) fulfill that constraint and

the Minkowski addition is linear with respect to support values so that $\mathbf{M}_\mathrm{C}\left(\sum_i \mathbf{h}_{S_i}\right) = \mathbf{0}$ necessarily holds. What remains is that $S_k + S_l$ fails to be representable in $\mathcal{H}_\mathrm{C}$-representation because of the matrix of facet normals $\mathbf{A}$. This is only possible when $S_k + S_l$ exhibits additional facets that are not covered with the original matrix of facet normals $\mathbf{A}$.

In summary, unified partitions have two equally valid interpretations. First, they distinguish different maximum a-types. Secondly, they separate different groups of structuring elements so that only structuring elements that are assigned to the same unified partition can be added to yield results in the original $\mathcal{H}_\mathrm{C}$-representation.

Literature. Borchert [2012] already solves a similar problem. In his work, he constructs an algorithm, similar to that of the Minkowski decomposition. Only the equality constraints are not considered. The corresponding cones in $\mathbf{h}_\mathrm{C}$-space are then used to assemble the validity cone partition by partition. Given a seed $\mathbf{h}_\mathrm{C}$-vector and the computed partition, new seed vectors outside of this cone are computed for each facet of the cone and in a predefined distance. This work of Borchert is strongly recommended for an alternative approach to the partitioning. The algorithm only contains two minor conceptual flaws[13]. First, it assumes that the constructed seed $\mathbf{h}_\mathrm{C}$-vectors cover all possible partitions. This is indeed very likely, but not guaranteed so that unified partitions might be missed out. Secondly, it assumes that all seed $\mathbf{h}_\mathrm{C}$-vectors represent a polytope with a maximum a-type which also cannot be guaranteed. In consequence, a wrong unified partition could be added that is not n_C-dimensional. Both situations are unlikely so that the algorithm by Borchert [2012] should most of the times properly compute the decomposition into unified partitions and the validity cone.

3.3.4 Conversion between $\mathcal{H}_\mathrm{C}$- and $\mathcal{S}$-representation

Motivation. With the preceding subsection, all structuring elements S_i are identified that are required to cover all polytopes of a $\mathcal{H}_\mathrm{C}$-representation. This subsection evaluates a mapping between the linear coefficients λ_i of the $\mathcal{S}$-representation and any valid $\mathbf{h}_\mathrm{C}$-vector to transform the measure equations that involve mixed volumes to the $\mathbf{h}_\mathrm{C}$-space.

Concept of mapping. Given an arbitrary valid $\mathbf{h}_\mathrm{C}$-vector, it can be assigned to at least one unified partition p. This partition is identified by the compliance of the $\mathbf{h}_\mathrm{C}$-vector with:

$$\mathbf{A}_{\mathrm{U},p}\mathbf{h}_\mathrm{C} \leq \mathbf{0}. \tag{3.25}$$

It follows that only the structuring elements associated with this unified partition can have coefficients $\lambda_i \neq 0$ for the desired $\mathcal{S}$-representation. Following subsection 2.4.2 (equation 2.55), for these structuring elements $S_i = P(\tilde{\mathbf{h}}_{\mathrm{C},S_i})$ holds:

$$P\left(\sum_i \lambda_i \tilde{\mathbf{h}}_{\mathrm{C},S_i}\right) = \sum_i \lambda_i S_i \tag{3.26}$$

[13] These issues are assumed to be hard to fix. They involve a numerical parameter (the distance of the new seeds outside of the known cones) whose magnitude has an upper bound that is influeced by the first issue and a lower bound that is influenced by the second issue. This comment is based on the experience that was gathered with previous versions of the decomposition algorithm that used a similar strategy. Given these experiences, the presented algorithm was designed to avoid numerical parameters.

so that (compare to equation 2.56 and multiply $\mathbf{M}_{\mathbf{h}\mapsto\mathbf{h}_\mathrm{C}}$ from the left):

$$\mathbf{h}_\mathrm{C} = \sum_i \lambda_i \tilde{\mathbf{h}}_{\mathrm{C},S_i}, \tag{3.27}$$

in principle, already provides the required mapping. However, a constrained Minkowski decomposition often identifies more than n_C structuring elements so that equation 3.27 cannot be inverted.

Simplex partitions. To obtain an invertible mapping, the selection of structuring elements must be reconsidered to allow only n_C structuring elements to have coefficients $\lambda_i \neq 0$. This can be obtained by decomposing the unified partition into cones with exactly n_C extreme rays. Therefore, the unified partitions $\mathcal{C}_{\mathrm{U},p}$ are closed and the corresponding polytopes $\bar{\mathcal{C}}_{\mathrm{U},p}$ are triangulated. Each resulting simplex is comprised of exactly n_C natively scaled extreme ray elements and the origin. Removing the closing hyperplane for all simplices, results in convex cones that are also a partitioning of the validity cone. These partitions will be called *simplex partitions*. They are denoted by $\mathcal{C}_{\mathrm{S},p}$, stored with the normal matrices $\mathbf{A}_{\mathrm{S},p}$ and their total number is given by n_PS.

Converting $\mathcal{H}_\mathrm{C}$- into $\mathcal{S}$-representation. Having a partitioning of the validity cone into simplex partitions, an appropriate partition for a vector $\mathbf{h}_\mathrm{C}$ is found when:

$$\mathbf{A}_{\mathrm{S},p}\mathbf{h}_\mathrm{C} \leq \mathbf{0} \tag{3.28}$$

is fulfilled and the corresponding structuring elements are denoted by S_{k_j} with n_C distinct indices $j \in \{1,\ldots,n_\mathrm{C}\}$. The coefficients λ_i are then given with:

$$(\lambda_{k_1},\ldots,\lambda_{k_{n_\mathrm{C}}})^T = (\mathbf{h}_{\mathrm{C},S_{k_1}},\ldots,\mathbf{h}_{\mathrm{C},S_{k_{n_\mathrm{C}}}})^{-1}\mathbf{h}_\mathrm{C} \tag{3.29}$$

$$\lambda_i = 0 \qquad i \notin \{k_1,\ldots,k_{n_\mathrm{C}}\}. \tag{3.30}$$

The matrix inversion $(\mathbf{h}_{\mathrm{C},S_{k_1}},\ldots,\mathbf{h}_{\mathrm{C},S_{k_{n_\mathrm{C}}}})^{-1}$ does not have to be calculated, given the following reasoning. A necessary condition for the coefficients is $\boldsymbol{\lambda} \geq \mathbf{0}$. If $\boldsymbol{\lambda}$ is substituted by equation 3.29, the inequality identifies all $\mathbf{h}_\mathrm{C}$-vectors that are covered with the corresponding structuring elements. This inequality, hence, equals the $\mathcal{H}$-representation of the simplex partition and it holds:

$$(\mathbf{h}_{\mathrm{C},S_{k_1}},\ldots,\mathbf{h}_{\mathrm{C},S_{k_{n_\mathrm{C}}}})^{-1} = -\mathbf{A}_{\mathrm{S},q} \tag{3.31}$$

Summary. The introduction of simplex partitions and the conversion from $\mathcal{H}_\mathrm{C}$-representation to $\mathcal{S}$-representation are important for the derivations of the following subsection. Despite of that, these results are not essential. Neither the decomposition into simplex partitions nor the mapping to a certain $\boldsymbol{\lambda}$-vector is unique so that neither the simplex partitions nor the $\mathcal{S}$-representation provide genuine information that is not already given by the unified partitions.

3.3.5 Measure Calculation

Motivation. Having a $\boldsymbol{\lambda}$-vector for each valid $\mathbf{h}_\mathrm{C}$-vector by the previous subsection, we can transfer the measure calculation by mixed volumes (see section 2.3.2) to the $\mathbf{h}_\mathrm{C}$-space. The derivations are performed for a 3-dimensional measure μ while the equations for one- and two-dimensional measures can be obtained analogously.

Derivation. From equations 2.32 and 3.29–3.31 it follows:

$$\mu(\mathbf{h}_\mathrm{C}) = -\sum_{i,j,k} \widetilde{V}_{(i,j,k)} \sum_{r,s,t} a_{\mathrm{S},p,(i,r)} a_{\mathrm{S},p,(j,s)} a_{\mathrm{S},p,(k,t)} h_{C,r} h_{C,s} h_{C,t}, \tag{3.32}$$

where $a'_{\mathrm{V},p,(i,r)}$ are the elements of the matrix $\mathbf{A}_{\mathrm{S},\,p}$ and $\widetilde{V}_{(i,j,k)} = \widetilde{V}(S_i, S_j, S_k)$ are mixed volumes. New coefficients can be defined by changing the order of summation:

$$\widehat{V}_{p,(r,s,t)} = -\sum_{i,j,k} a_{\mathrm{S},p,(i,r)} a_{\mathrm{S},p,(j,s)} a_{\mathrm{S},p,(k,t)} \widetilde{V}_{(i,j,k)} \tag{3.33}$$

and the measure computation simplifies to:

$$\mu(\mathbf{h}_\mathrm{C}) = \sum_{i,j,k} \widehat{V}_{p,(i,j,k)}\, h_{\mathrm{C},i} h_{\mathrm{C},j} h_{\mathrm{C},k} \tag{3.34}$$

where the indices r, s, t are substituted back to i, j, k.

Normalized coefficients. The coefficients $\widehat{V}_{p,(i,j,k)}$ are not uniquely determined for the purpose of calculating the desired measure. As it is summed over all combinations i, j and k the same product $h_{\mathrm{C},i} h_{\mathrm{C},j} h_{\mathrm{C},k}$ appears several times. For example, the result of equation 3.34 remains unchanged using $\left(\widehat{V}_{p,(1,2,3)} + 1\right)$ and $\left(\widehat{V}_{p,(1,3,2)} - 1\right)$ instead of $\widehat{V}_{p,(1,2,3)}$ and $\widehat{V}_{p,(1,3,2)}$. Uniquely determined coefficients are obtained when the index combinations are restricted to $i \geq j \geq k$ so that every product $h_{\mathrm{C},i} h_{\mathrm{C},j} h_{\mathrm{C},k}$ appears only once. This defines the following normalized coefficients:

$$\widehat{V}'_{p,(i,j,k)} = \begin{cases} \widehat{V}_{p,(i,j,k)} + \widehat{V}_{p,(i,k,j)} + \widehat{V}_{p,(j,i,k)} & \\ +\widehat{V}_{p,(j,k,i)} + \widehat{V}_{p,(k,i,j)} + \widehat{V}_{p,(k,j,i)} & i > j > k \\ \widehat{V}_{p,(i,j,k)} + \widehat{V}_{p,(j,i,k)} + \widehat{V}_{p,(j,k,i)} & i > j \text{ and } j = k \\ \widehat{V}_{p,(i,j,k)} + \widehat{V}_{p,(i,k,j)} + \widehat{V}_{p,(k,i,j)} & i = j \text{ and } j < k \\ \widehat{V}_{p,(i,j,k)} & i = j = k \\ 0 & i < j \text{ or } j < k \end{cases}. \tag{3.35}$$

Elimination of simplex partitions. The derivation suggests that the simplex partitions are required to determine the set of coefficients $\widehat{V}'_{p,\,(\cdot)}$ as the $\mathbf{h}_\mathrm{C}$-vector is assumed to belong to a specific simplex partition. This is not true. It is known that equation 3.34 holds for any $\mathbf{h}_\mathrm{C}$-vector of the same a-type (see subsection 2.3.1). Hence, sets of normalized coefficients must be equal when the corresponding simplex partitions belong to the same unified partition. In conclusion, the sets of normalized coefficients can be summarized to one set for each unified partition which will be denoted by $c_{p,(i,j,k)}$. Coefficients for an unspecified measure will be denoted by $c_{p,(\cdot)}$ as the dimensionality of this measure is unknown and it may be $c_{p,(i,j,k)}$, $c_{p,(i,j)}$ or $c_{p,\,(i)}$. If a specific measure is addressed, the coefficients will be named accordingly, like $c^{\text{volume}}_{p,(i,j,k)}$ or $c^{\text{surface}}_{p,(i,j)}$.

Summary. Given a $\mathcal{H}_\mathrm{C}$-representation, the validity cone can be calculated and decomposed into unified partitions. The mixed volumes for any measure calculation are calculated according to subsection 2.3.2 and the corresponding measure equations are transferred to the $\mathbf{h}_\mathrm{C}$-space according to this subsection. All these steps can be executed

a priori to any application so that measure calculation reduces to two steps. First, an appropriate unified partition is determined by equation 3.25. Secondly, the measure is determined with:

$$\mu_{p,(i,j,k)} = \sum_{i,j,k} h_{\mathrm{C},i} h_{\mathrm{C},j} h_{\mathrm{C},k} c_{p,(i,j,k)}. \tag{3.36}$$

Remark. In contrast to the discussion in subsection 2.3.1, the constraint is relaxed that polytopes must have the same a-type in order to share the same set of coefficients for measure computation. Instead, the above equation is valid for any $\mathbf{h}_{\mathrm{C}}$-vector of a unified partition and, hence, for any a-type $[P] \leq [Q]$ where $[Q]$ is the maximum a-type associated with the unified partition. Though, the coefficients $c_{p,(i,j,k)}$ are still dependent on the maximum a-type.

3.4 Case Study and Summary

3.4.1 Introduction

Motivation. In the discussion of section 3.2, proper $\mathcal{H}_C$-representations are isolated from general $\mathcal{H}_C$-representations. They efficiently handle the disappearance and reformation of facets and measure calculations. However, the discussion in the preceding sections was based on theory, only. In practise, limits exist that are connected to the underlying combinatorial complexity of the geometry (*e.g.* the number of possible a-types) and the corresponding computation times. These limitations are explored in this subsection based on a large set of test cases. Additionally, the algorithms utilize several polytope operations like the conversion between $\mathcal{V}$- and $\mathcal{H}$-representation (see appendix A.1.2), volume computation, set intersection or Minkowski decomposition. While the algorithms work well in theory, their practical implementation must consider numerical precission errors, a corresponding scaling and tolerances. The presented case study is, hence, simultaneously used to test the implementation.

Content. To evaluate the geometric complexity and corresponding computation times [Desktop Computer, 2009], every detail from sections 3.2 and 3.3 is computed which includes the validity cone, the unified and simplex partitions, the structuring elements and the sets of coefficients for measure computation. To verify these results, several consistency checks are implemented that verify, for example, that the whole validity cone is covered by unified partitions or that the normalized coefficients for measure computation ($c_{p,(i,j,k)}$) are indeed identical for each simplex partition of the same unified partition. Additionally, several sample $\mathbf{h}_C$-vectors are generated to verify the correctness of measure computation by the presented framework. These tests are also used to evaluate the average time for measure computations.

Cases. The considered cases are generated from four different crystal systems that are: potash alum, paracetamol, magnesium sulfate undecahydrate ($MgSO_4 \cdot 11H_2O$) and α-lactose monohydrate (lactose). The crystals are shown in the top row of figure 3.7 in a typical appearance while the symmetry information and lattice parameters are summarized in table 3.1. The bottom row of figure 3.7 additionally shows four more complex cases that are computed. The present crystal forms for each case are taken from a set of obligatory forms that ensures a closed polyhedron and a subset of optional forms. The optional forms are binarily encoded and the resulting cases are listed by their corresponding decimal number in the right column of table 3.1. An example illustrates this encoding. Potash alum uses the obligatory form $\{111\}$ which might be augmented by the forms $\{100\}$, $\{110\}$, $\{210\}$ and/or $\{120\}$. Each form is assigned to a bit of the binary number (*e.g.* $\{100\}$ to 2^0 and $\{120\}$ to 2^3) so that the decimal number $11 = 1011_\mathrm{b}$ uses the forms $\{111\}$, $\{100\}$, $\{110\}$ and $\{120\}$ but not $\{210\}$. A summary of the realized complexity in terms of the dimension n_C is additionally provided by table 3.2.[14]

Measures. To evaluate the computation of measures, the volume, surface area and four Feret diameters are considered. For the Feret diameters, one direction is chosen as $\mathbf{u}_1 = (0, 0, 1)^T$ and the other three are randomly generated with: $\mathbf{u}_2 = (0.4966,$

[14] For lactose, the position is not fixed based on the crystal forms or symmetry. Table 3.2 provides the dimension n_C after the position is fixed according to subsection 3.2.1.

Table 3.1.: Considered crystal systems and cases

crystal and crystal structure	obligatory faces	optional faces	considered combinations of forms
potash alum			
cubic, $m\bar{3}$, $a = 12.158\,\text{Å}$, [Ma et al., 2008]	$\{111\}$	$\{100\}$, $\{110\}$, $\{210\}$, $\{120\}$	0-15
paracetamol			
monoclinic, $2/m$, $a = 12.651\,\text{Å}$, $b = 8.887\,\text{Å}$, $c = 7.236\,\text{Å}$, $\beta = 114.848°$, [Borchert et al., 2009]	$\{110\}$, $\{011\}$, $\{001\}$	$\{20\bar{1}\}$, $\{101\}$, $\{100\}$, $\{010\}$, $\{111\}$, $\{\bar{1}11\}$, $\{\bar{1}01\}$	0-16, 19, 21, 23, 27, 31-32, 64
$MgSO_4 \cdot 11H_2O$			
triclinic, $\bar{1}$, $a = 6.72548\,\text{Å}$, $b = 6.77937\,\text{Å}$, $c = 17.2898\,\text{Å}$, $\alpha = 88.255°$, $\beta = 89.478°$, $\gamma = 62.598°$, [Genceli et al., 2007]	$\{100\}$, $\{010\}$, $\{001\}$	$\{110\}$, $\{1\bar{1}0\}$, $\{011\}$, $\{0\bar{1}1\}$, $\{101\}$, $\{\bar{1}01\}$, $\{111\}$, $\{11\bar{1}\}$, $\{1\bar{1}1\}$, $\{1\bar{1}\bar{1}\}$	0-21, 23-32, 35, 37-39, 41-43, 45-47, 51, 53-55, 57-59, 61-62, 64, 79, 87, 95, 128, 256, 512
lactose			
monoclinic, 2, $a = 0.7982\,\text{Å}$, $b = 2.1563\,\text{Å}$, $c = 0.4824\,\text{Å}$, $\beta = 109.57$, [Fries et al., 1971]	$\{010\}$, $\{0\bar{1}0\}$, $\{100\}$, $\{0\bar{1}\bar{1}\}$	$\{110\}$, $\{\bar{1}\bar{1}0\}$, $\{01\bar{1}\}$, $\{001\}$, $\{111\}$; $\{11\bar{1}\}$, $\{\bar{1}\bar{1}\bar{1}\}$, $\{1\bar{1}\bar{1}\}$	0-16, 19, 21, 23, 27, 31-32, 64, 128

Table 3.2.: Number of cases for each crystal system separated by the number of facet groups n_C

number of face groups	1	2	3	4	5	6	7	8	9
potash alum	1	4	6	4	1				
paracetamol			1	7	6	6	3	1	
$MgSO_4 \cdot 11H_2O$			1	10	10	14	14	8	1
lactose			1	8	6	6	3	1	

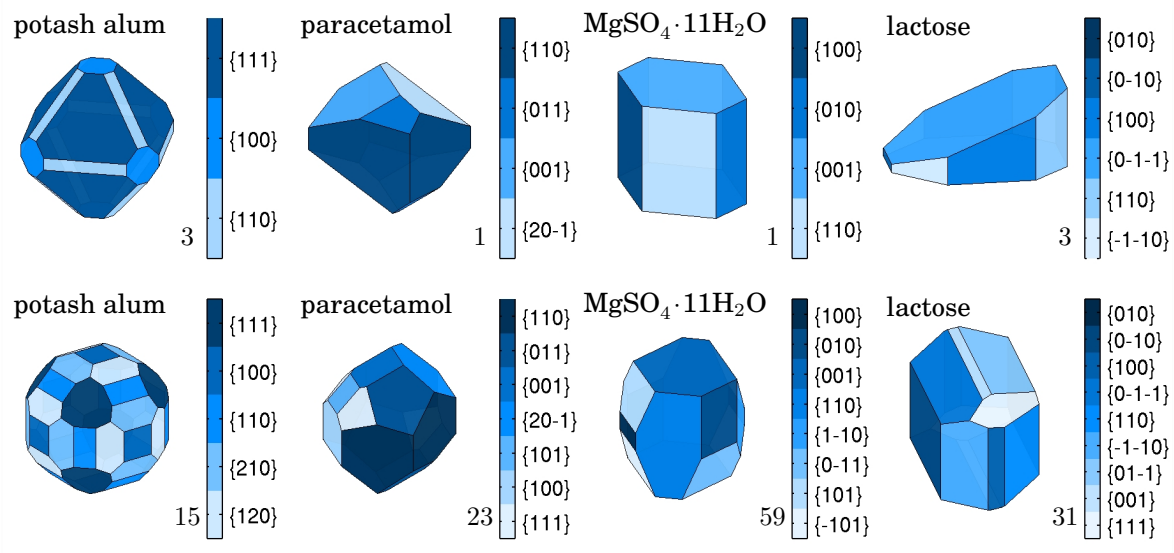

Figure 3.7.: Sample crystals from the case study (the number left to the colorbar indicates the combination of forms from table 3.1)

$0.6449, 0.3420)^T$, $\mathbf{u}_3 = (0.8998, 0.8180, 0.2897)^T$ and $\mathbf{u}_4 = (0.8216, 0.6602, 0.3412)^T$. Measure computation is too fast so that, typically, multiple sample points must be evaluated to obtain a reasonable accurate time measurement. Therefore, the presented results in the following subsection are averaged computation times.

3.4.2 Results and Discussion

Combinatorial complexity. The number of unified partitions in dependence of the dimensionality of the $\mathbf{h}_\mathrm{C}$-space is shown in the left of figure 3.8. While a roughly exponential dependency can be observed, the obtained number of unified partitions is widely scattered. The most complex example in the case study generated almost 1000 unified partitions which correspond to 1000 maximum a-types. The number of structuring elements in dependence of the dimension n_C is shown in the right of figure 3.8. Again, the dependency is exponential while the data is less scattered. Several more complex cases use about 100 structuring elements.

***A priory* computation times.** Over 90% of the total computation time to generate all details from sections 3.2 and 3.3 originates from three separate steps. The first time consuming step is the generation of the inequalities for the validity cone $\mathcal{C}'_\mathrm{V}$ in $\mathbf{h}$-space (see subsection 2.4.3). This time is shown in the left of figure 3.9 and is proportional to n_H^4 since the algorithm iterates over quadrupels of facets. The largest computation time is given by 36 seconds. The second time consuming step is the decomposition of the validity cone $\mathcal{C}_\mathrm{V}$ into unified partitions $\mathcal{C}_{\mathrm{U},p}$. A roughly linear dependency on the number of unified partitions can be observed in the right of figure 3.10. This is reasonable since each unified partition requires the execution of one constrained Minkowski decomposition and one set difference. The largest computation time for the decomposition of the validity cone is given by 3600 seconds or one hour. In both steps, the sample points for potash alum are clearly separated from the points of other cases. This is caused by

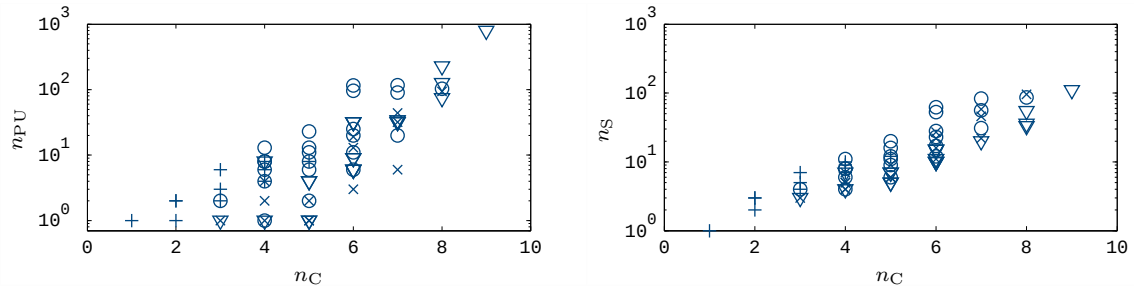

Figure 3.8.: Number of unified partitions (left) and number of structuring elements (right) plotted versus the number of independent facet groups; the crystal systems are indicated by (+) for potash alum, (◯) for paracetamol, (▽) for $MgSO_4 \cdot 11H_2O$, and (×) for lactose

the high symmetry of the potash alum system that contains more faces per facet group than the other crystal systems while the generation of the inequalities for the validity cone and for the Minkowski decomposition iterate over quadrupels of facets. The third time consuming step is the computation of the mixed volumes $\tilde{V}$ for the volume, surface area and Feret diameters which are shown in figure 3.10 in dependence of the number of structuring elements. If all possible mixed volumes are computed, these times would be proportional to n_{S}^n where n is the dimensionality of the measure. Corresponding straight lines are added to figure 3.10.[15] However, a mixed volume $\tilde{V}(S_i, S_j, S_k)$ is only required when the structuring elements S_i, S_j and S_k belong to a common unified partition (see subsection 3.3.5). The implementation considers this detail and skips unnecessary mixed volumes so that considerable time savings are visible for $n = 2$ and $n = 3$.

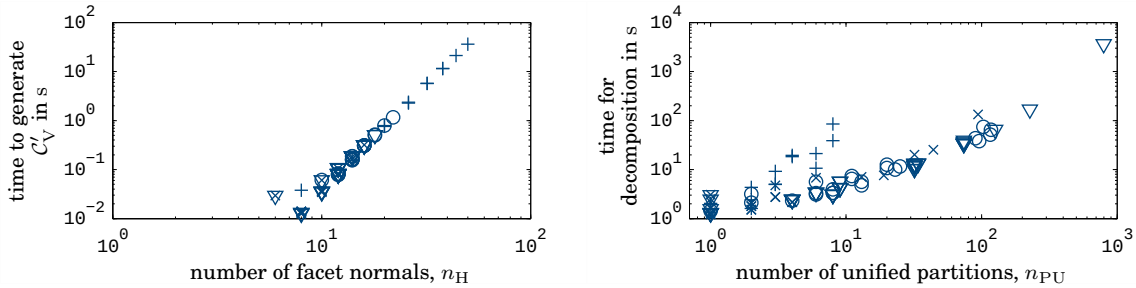

Figure 3.9.: Computation times for generating the inequalities of the unified partition (left) and for the decomposition of the validity cone into unified partitions (right); the crystal systems are indicated by (+) for potash alum, (◯) for paracetamol, (▽) for $MgSO_4 \cdot 11H_2O$, and (×) for lactose

Times for measure computation. Figure 3.11 demonstrates the computation times that are necessary to determine the volume based on a given $\mathbf{h}_{\mathrm{C}}$-vector. For the derived framework, the measure computation comprises two steps. First, the unified partition of a given $\mathbf{h}_{\mathrm{C}}$-vector must be identified and the required times are shown by crosses

[15] These lines are not fitted and drawn manually.

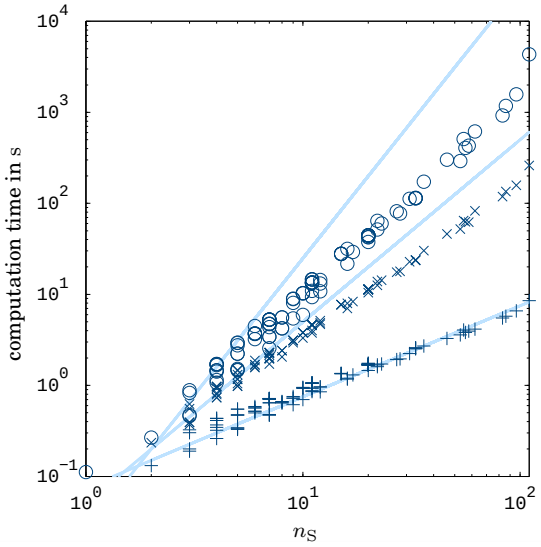

Figure 3.10.: Computation times for the generation of the mixed volumes $\tilde{V}_{(\cdot)}$ where (◯) indicates the generation for the crystal volume, (×) for the crystal surface area and (+) for a Feret diameter; the straight lines indicate functions proportional to n_S^1, n_S^2 and n_S^3

in figure 3.11. Since the appropriate unified partition is identified by subsequently checking the containment in the partition (equation 3.25), this time is proportional to the number of unified partitions n_{PU}. In the second step, the volume is evaluated based on a 3^{rd} order polynomial dependent on the components of the $\mathbf{h}_C$-vector. The required computation time for this step is much smaller, shown by circles in figure 3.11. In comparison, the computation of the volume based directly on the $\mathcal{H}$-representation by the quick hull algorithm results in an almost constant timing of $62\,\mathrm{ms}$ denoted by the + symbols.

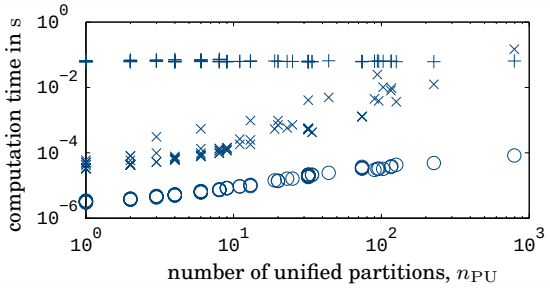

Figure 3.11.: Computation times to determine the unified partition of a $\mathbf{h}_C$-vector (×), to determine the measure based on a $\mathbf{h}_C$-vector and known unified partition (◯) and computation times to compute the measures based only on the $\mathcal{H}$-representation (+)

3.4.3 Conclusions and Summary

Geometric complexity. The case study considered crystal systems with up to 9 independently growing facet groups. Since the number of unified partitions grows exponentially with n_C, the computational effort increases likewise. It follows that proper $\mathcal{H}_\mathrm{C}$-representations with a dimension of n_C in the order of 10 must be considered being high dimensional. On the other hand, such high dimensional crystal systems are not reasonable in applications so that typically up to a couple of minutes is required to provide the data for validity mapping and efficient measure computations.

Measure computations. The most apparent benefit of the provided framework is given by the efficiency of measure computations. The appropriate unified partition (and a-type) can be determined by a set of linear inequalities (equation 3.25) while the measure computation evaluates a 3^{rd} or lower order polynomial (equation 3.36). Corresponding time savings compared to the computation of the same measures directly from the $\mathcal{H}$-representation are crucial for the evaluation of population balances (closing of solute mass balances) as it is shown by Borchert [2012] or subsequent chapters of this work.

Understanding of the geometric state space. A much more subtle achievement of this chapter is the general understanding of the geometric state space. The unified partitions and their boundaries indicate all possible a-types or morphologies of a crystal system so that any morpholgy related question can be addressed efficiently. Analogously, the validity cone and the facet validity cones indicate regions in $\mathbf{h}_\mathrm{C}$-space where certain facets are present. Based on this information, it is straightforward to determine if facets grow out, given a particular growth rate. Even though these results are already evident by Borchert [2012], this work comprises two differences. First, the framework is connected to existing mathematic theory from which Minkowski addition is the most important concept. This connection allows a seamless extension to rounded particles in chapter 5 while rounded particles that include spheres cannot even be expressed by a $\mathcal{H}$-representation. Secondly, the derivation considers a generalization to arbitrary mapping matrices $\mathrm{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}$ from which the distinction between proper and improper $\mathcal{H}_\mathrm{C}$-representations emerged. Improper representations are used in chapter 6 for shape approximation and model reduction.

Validity mapping. The algorithm for the correction of $\mathbf{h}_\mathrm{C}$-vectors with invalid facet distances (validity mapping) is probably the most important achievement of this chapter. The principles of this algorithm are utilized in chapter 4 to reduce the stiffness of the numerical scheme that is used to solve population balances. But the concept of validity mapping is even more important for chapter 6 where two approaches for shape approximation are presented and improper representations are analyzed.

4

Dynamic Evolution of Crystals

4.1 Introduction and Preliminaries

4.1.1 Introduction

Motivation. Chapter 3 introduces a rigorous framework to model the geometry of faceted crystals. While the application of $\mathcal{H}_C$-representations is motivated by crystal growth, the resulting dynamic evolution of single crystals or crystal populations is not discussed there. These details are considered in this chapter separately, because we change the viewpoint from derivation to application for the geometrical framework.

Single crystals. Considering growth and dissolution, the evolution of a crystal population is governed by the dynamics of each involved crystal. These dynamics are described by a time dependent growth (or dissolution) rate $\tilde{\mathbf{g}}(t)$:

$$\frac{\mathrm{d}\mathbf{h}_C}{\mathrm{d}t} = \tilde{\mathbf{g}}(\mathbf{h}_C, t) \tag{4.1}$$

while an initial crystal shape $\mathbf{h}_C(0)$ is assumed. Because every vector $\mathbf{h}_C(t)$ must remain valid, this general formulation must obey the bounds of the validity cone so that $\tilde{\mathbf{g}}$ is necessarily shape dependent. The original growth rate given by kinetic laws does typically not consider the boundaries of the validity cone and is denoted by $\mathbf{g}$. At this point, recall that the boundary of the validity cone represents $\mathbf{h}_C$-vectors for which some facets have grown out. In literature, the typical approach to comply with these bounds is to substitute components of the growth rate when corresponding facets have grown out and do not reappear. They are substituted by a rate that corresponds to the velocity of the supporting vertex for that facet [Zhang et al., 2006, Borchert et al., 2009, Borchert, 2012, Singh, 2013]. The method in this work uses a more abstract approach since it is derived from the more general projections of the validity mapping (see section 3.2). This method is presented in subsection 4.2.1. The difference between both approaches is very subtle because it lies within the derivation, not within the immediate results. However, the conducted approach also addresses and resolves stiffness issues that appear for population balances and which are not treated in known literature.

Crystal populations. A population of particles is commonly described by distribution densities $\tilde{n}$ with respect to spatial and/or intrinsic coordinates. In the scope of this thesis, well mixed batch reactors are assumed and the crystal population $\tilde{n}$ is distributed

over the coordinates of a $\mathcal{H}_\mathrm{C}$-representation so that:

$$I_1 = \int \tilde{n}(\mathbf{h}_\mathrm{C})\,\mathrm{d}\mathbf{h}_\mathrm{C} \tag{4.2}$$

represents the total number of particles. Deriving a balance equation for the distribution density $\tilde{n}(\mathbf{h}_\mathrm{C},\,t)$ results in the population balance equation together with initial, boundary and regularity conditions ([Randolph and Larson, 1971], [Ramkrishna, 2000, equations 2.7.9 to 2.7.11], [Borchert, 2012, equations 2.58a to 2.58d]):

$$\begin{aligned}
\frac{\partial \tilde{n}}{\partial t} + \nabla_{\mathbf{h}_\mathrm{C}} \bullet (\tilde{\mathbf{g}}\,\tilde{n}) &= \sum_i \sigma_i, && (4.3)\\
\tilde{n}(\mathbf{h}_\mathrm{C},\,0) &= \tilde{n}_0(\mathbf{h}_\mathrm{C}), && (4.4)\\
\mathbf{A}_\mathrm{V}\left(\tilde{\mathbf{g}}(\mathbf{h}_\mathrm{C},\,t)\,\tilde{n}(\mathbf{h}_\mathrm{C},\,t)\right) &= \mathbf{0} \qquad \text{for } \mathbf{h}_\mathrm{C} \in \partial\mathcal{C}_\mathrm{V}, && (4.5)\\
\tilde{n}(\mathbf{h}_\mathrm{C},\,t) &\to 0 \qquad \text{for } \|\mathbf{h}_\mathrm{C}\| \to \infty, && (4.6)
\end{aligned}$$

where $\nabla_{\mathbf{h}_\mathrm{C}}$ is the Nabla operator so that $\nabla_{\mathbf{h}_\mathrm{C}} \bullet (\tilde{\mathbf{g}}\,\tilde{n})$ represents the divergence of the vector field $(\tilde{\mathbf{g}}\,\tilde{n})$. Note that equation 4.3 usually does not contain any source or sink term σ_i for the scope of this work because none of the corresponding mechanisms are relevant. The shapes of aggregated or broken particles are beyond the scope of the geometrical framework. Furthermore, without aggregation or breakage, nucleated particles can be separated as a 1-dimensional population balance [Borchert, 2012] for which numerous approaches are available [Kumar and Ramkrishna, 1997, Briesen, 2009b, Borchert, 2012]. What remains is the evolution of an initial crystal population $\tilde{n}_0(\mathbf{h}_\mathrm{C})$ by growth and dissolution which is still challenging as it will be discussed in the following subsection. Dissolution differs from growth by an additional sink term σ_i that removes dissolved particles from the system.

4.1.2 Solving Population Balances

Sources of complexity. The focus of this chapter is a relatively simple crystallization problem in terms of included mechanisms, though is it mathematically classified as a hyperbolic partial differential equation and thereby considered to be difficult to solve. Efficient numerical schemes are challenging and analytic solutions only exist for simplified cases. Beyond that, the population balance (equation 4.3) does not pose the complete system of equations. Considering a real crystallization process, it involves a growth rate that depends on supersaturation and, hence, requires the closure of the mass balance. Therefore, it is required to evaluate the total volume of the particle population:

$$I_\mathrm{volume}(t) = \int \mu^\mathrm{volume}(\mathbf{h}_\mathrm{C})\,\tilde{n}(\mathbf{h}_\mathrm{C},\,t)\,\mathrm{d}\mathbf{h}_\mathrm{C}. \tag{4.7}$$

The numerical computation of this integral becomes more and more challenging with increasing dimension n_C so that a numerical scheme for the population balance must also consider a numerical scheme for this integral.

The second source for the complexity of the problem is inherited from the geometrical complexity of the single crystal model which is mainly posed by the validity cone. Details for the proposed numerical scheme in subsection 4.2.2 that also apply for the moving pivot technique are given in subsection 4.2.1.

Literature overview. Numerical schemes to solve the population balance can be subdivided into different classes for which an extended overview is available by Gunawan et al. [2004], Costa et al. [2007] or Mesbah et al. [2009]. The discussed classes of numerical schemes contain Monte Carlo simulation, method of moments, finite differences such as the Lax-Wendroff method and weighted residuals such as the finite elements method. However, the most successful class of methods for the focused problem seems to be formed by discretization techniques, in particular finite volume high resolution schemes [Leveque, 2002, Ma et al., 2002] and the moving pivot technique [Kumar and Ramkrishna, 1997, Borchert and Sundmacher, 2011, Borchert, 2012]. The method of characteristics will also be discussed but is considered rather an analytical method than a numerical scheme.

Finite volumes. When the integral form of the population balance is applied for closed subdomains of the crystal state space, the resulting methods are classified as finite volume schemes.

First order schemes of this kind typically suffer from numerical diffusion. Ma et al. [2008] used the method of classes to model shape-independent growth in 3 dimensions for potash alum. The boundary conditions are not considered, even though, at the end of the presented simulation, large parts of the particle population are outside of the validity cone. Shortly after, the model is extended to shape-dependent growth while the geometrical boundary conditions remain unconsidered [Ma and Wang, 2008].

High resolution schemes prevent numerical diffusion, a typical disadvantage of first order methods, as well as oscillations, a typical disadvantage of finite difference methods. Ma et al. [2002] demonstrate how the high resolution scheme can be viewed as a hybrid of the upwind scheme (first order finite volumes) and the Lax-Wendroff method (second order finite differences). In their study, growth and nucleation in 2 dimensions are considered for potassium dihydrogen phosphate (KDP). The restrictions of the validity cone are not considered while in this case the population remains in the validity cone. Briesen [2009a] also applies the high resolution scheme and models attrition as growth in 2 dimensions. The derived growth rates natively comply with the resulting boundary conditions. This particular study is discussed in more detail in chapter 5.

Typical for all approaches above are regular grids so that the given numerical schemes scale poorly into n dimensions. If the grid spacing and, hence, the accuracy is maintained, the computational load grows exponentially with the dimension. Using irregular grids is, in principle, possible but such numerical schemes are not yet tested for the solution of population balance equations [Leveque, 2002].

Pivots. A very popular approach approximates the distribution density with a finite sum of Dirac delta functions for which each delta function marks a pivot point in crystal state space. The fixed pivot technique, introduced by Kumar and Ramkrishna [1996], never changes the positions of the initially selected pivots. While this method is already extended for n dimensions by Chakraborty and Kumar [2007], it suffers from numerical diffusion for most growth applications. In the moving pivot technique, the initially selected pivots move according to the growth rate so that numerical diffusion is eliminated. It is introduced by Kumar and Ramkrishna [1997] and extended to n dimensions by Borchert [2012]. The problem, demonstrated by Borchert [2012], models 2-dimensional growth for potassium dihydrogen phosphate (KDP).

For pivot techniques, the evaluation of the total particle volume, equation 4.7, is triv-

ial but the problem of n-dimensional integration is shifted to the realization of the initial condition. Borchert uses randomly sampled pivots that are regularized in a second step. To assign particle numbers to each pivot, the initial condition is integrated over the cells of a Voronoi tessellation. Thereby, the shape of each integration domain results in a poor efficiency of general purpose integrators [Hahn, 2005]: an increasing dimension of the problem leads to a decay of the volume fraction between a Voronoi cell and its rectangular bounding box[1]. Additionally, a high number of integrations must be computed when a high number of pivots is chosen. In summary, the integrals for the Voronoi cells cannot be solved effectively in n dimensions. While a scaling of the problem or a derivation of a tailored integration scheme is possible, the complexity of that task is assumed to be very high.

Method of characteristics. An approach that is similar to the moving pivot technique can be derived based on the method of characteristics which is not a numerical scheme but an approach to solve partial differential equations (semi-)analytically. Therefore, the total differential:

$$\frac{\mathrm{d}\tilde{n}(\mathbf{h}_\mathrm{C}(t),\, t)}{\mathrm{d}t} = \frac{\partial \tilde{n}}{\partial t} + (\nabla_{\mathbf{h}_\mathrm{C}} \tilde{n}) \bullet \frac{\mathrm{d}\mathbf{h}_\mathrm{C}}{\mathrm{d}t} \tag{4.8}$$

is utilized and the partial time derivative of the distribution density is substituted with the population balance, equation 4.3, to obtain:

$$\frac{\mathrm{d}\tilde{n}(\mathbf{h}_\mathrm{C}(t),\, t)}{\mathrm{d}t} = -\tilde{n}\,(\nabla_{\mathbf{h}_\mathrm{C}} \bullet \tilde{\mathbf{g}}) + \sum_i \sigma_i. \tag{4.9}$$

The resulting ordinary differential equation gives the distribution density at a position $\mathbf{h}_\mathrm{C}(t)$ that continuously moves according to the growth rate (equation 4.1). A numerical scheme is obtained, when a finite number of sampling points $\mathbf{h}_{\mathrm{C},i}(0)$ is selected for which the evolution of the initial condition is tracked and when also a numerical scheme is given to evaluate the total particle volume (equation 4.7). In contrast to the moving pivot technique, the pivots do not represent a specific number of particles. They indicate the distribution density at this point. In consequence, the integration of the total volume, equation 4.7, requires additional considerations which might pose assumptions to the distribution of the pivots in $\mathbf{h}_\mathrm{C}$-space. Therefore, the method of characteristics does not constitute a complete numerical scheme for most process models.

Singh and Ramkrishna [2013] propose this scheme for problems with an equivalent geometrical framework, but they neither specify the discretization scheme, nor do they specify the integration algorithm for equation 4.7.

Monte Carlo Simulations. Monte Carlo methods simulate a large number of individual particles [Ramkrishna, 1981, Smith and Matsoukas, 1998]. The initial population is generated randomly and mechanisms like aggregation or breakage are modeled

[1] Imagine an n-dimensional ball which is a very regular body compared to the irregular Voronoi cells. The values $n = 1$, $n = 2$, $n = 3$ should be sufficient for the general principle. The volume fraction between these balls and their surrounding rectangular bounding boxes decays with increasing dimension n from 1 to $\frac{pi}{4} \approx 0.79$ to $\frac{pi}{6} \approx 0.52$. Hence, a general purpose solver will have problems to identify the region of interest for integration (the ball of Voronoi cell) inside the rectangular bounding box. Even though problems are considered to be high-dimensional for $n \approx 100$, the shape of Voronoi cells can be particularly undesireable.

by stochastic events. The size or shape distribution is then obtained by a statistical evaluation. The benefit of this approach is that complex kinetic mechanisms can be introduced easily. The disadvantage is a relatively high computational cost since large numbers of particles must be simulated. Note that even though Monte Carlo integral estimates are used as the basis for the numerical scheme in this work, the derived method is very different from Monte Carlo simulations.

Summary. From the present state of literature, the only scheme that might scale well into n dimensions without major extensions is given by Borchert [2012]. Inferentially, this approach was tested while severe numerical problems occurred for high numbers of pivots (obtaining a proper Voronoi tessellation) or an increasing dimension of the problem so that scaling or tailored integrators seem to be required. However, during the analysis of these problems, a different approach to derive a numerical scheme emerged that is presented in subsection 4.2.2. Despite of Monte Carlo simulations, all numerical schemes mentioned in the beginning are derived focusing on the challenges of the population balance. In the new scheme, the derivation concentrates on a numerical scheme for the evaluation of the total volume.

4.2 Numerical Considerations

4.2.1 Geometrical boundaries

Single crystals. Figure 4.1 illustrates why growth rates cannot be chosen arbitrarily when the state vector must remain valid during the growth of a single crystal. Here, the crystal state $\mathbf{h}_\mathrm{C}$ is just valid, residing at the boundary of the validity cone $\mathcal{C}_\mathrm{V}$. Simultaneously, the growth rate points outside of the cone so that the evolution of that crystal state would lead to an invalid $\mathbf{h}_\mathrm{C}$-vector.

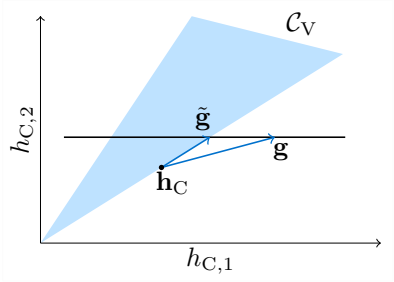

Figure 4.1.: Mapping growth rates to maintain valid $\mathbf{h}_\mathrm{C}$-vectors during growth.

Given the validity mapping for arbitrary $\mathbf{h}_\mathrm{C}$-vectors, described in subsection 3.2.3, the correction of growth rates $\mathbf{g}$ dependent on the current geometrical state $\mathbf{h}_\mathrm{C}$ is straight forward. Every boundary i of the facet validity cones $\mathcal{C}_{\mathrm{FV}(k)}$ can be treated sequentially, which results in steps:

$$\mathbf{g}' = \mathbf{P}_{\mathrm{FV}(k),i}\mathbf{g} \qquad \text{for } \mathbf{a}^T_{\mathrm{V}(k),i}\mathbf{h}_\mathrm{C} \geq 0 \text{ and } \mathbf{a}^T_{\mathrm{FV}(k),i}\mathbf{g} > 0 \tag{4.10}$$

while the whole procedure can be summarized as:

$$\tilde{\mathbf{g}} = \left(\prod_{(k,i)} \mathbf{P}_{\mathrm{FV}(k),i} \right) \mathbf{g} \tag{4.11}$$

analogously to equation 3.19.

Population balances. The geometrical constraints are included in the population balance via the boundary condition, equation 4.5, which is fulfilled whenever the corrected growth rates $\tilde{\mathbf{g}}$ are applied. However, the sharp gradient of the growth rate $\tilde{\mathbf{g}}$ results in a stiff system for the moving pivot technique or the technique derived in the following subsection. In one approach, Borchert [2012] avoids this problem by treating pivots at the boundary of the validity cone as separate populations. Hence, instead of adopting the growth rate $\mathbf{g}$ to fulfill the boundary condition, he adopts the number density $\tilde{n}$ by adding appropriate sink terms to the population balance. In consequence, the problem of having a stiff system of ordinary differential equations is transformed to the problem of tracking discrete events which can be considered equally complicated.

In this work, the problem of stiffness is handled by introducing a small margin region in the interior of the validity cone:

$$-\delta \cdot \mathbf{1} \leq \mathbf{A}_\mathrm{V}\mathbf{h}_\mathrm{C} \leq \mathbf{0} \tag{4.12}$$

for which the growth rate correction gradually sets in. Based on this margin region, the procedure for a single step (equation 4.10) is adopted to[2]:

$$\begin{aligned} \mathbf{g}' &= \left((1-\gamma)\,\mathbf{I}_{n_\mathrm{C}} + \gamma\,\mathbf{P}_{\mathrm{FV}(k),i}\right)\mathbf{g} \\ &\text{with } \gamma = \left(\frac{\mathbf{a}_{\mathrm{FV}(k),i}^T\mathbf{h}_\mathrm{C} + \delta}{\delta}\right) \\ &\text{for } \mathbf{a}_{\mathrm{V}(k),i}^T\mathbf{h}_\mathrm{C} \geq -\delta \text{ and } \mathbf{a}_{\mathrm{V}(k),i}^T\mathbf{g} > 0. \end{aligned} \tag{4.13}$$

Because each boundary i of a facet validity cone k identifies a specific vertex at which the facet will disappear, the parameter δ represents the distance between the facet and the corresponding vertex before the growth rate correction sets in. This approach to bypass stiffness implies that the boundary of the validity cone is never reached in finite time.

Dissolution. While the above procedure is motivated by growth, the same procedure is applicable to dissolution. However, the boundary condition of the population balance should in this case be satisfied by $\tilde{n} = 0$. This requirement can be implemented easily for the moving pivot technique or the technique introduced in the following subsection. Pivots that represent dissolved crystals can be removed from the system when:

$$\mathbf{a}_{\mathrm{FV}(0),i}^T\mathbf{h}_\mathrm{C} > -\delta \tag{4.14}$$

holds for any boundary i of the facet validity cone that identifies nonempty polytopes. Following this approach the regions in crystal state space at which dissolution occurs are directly identified by $\mathcal{C}_\mathrm{V} \cap \partial\mathcal{C}_{\mathrm{FV}(0)}$ which always includes the origin $\mathbf{0}$.

4.2.2 Solving Seeded Growth Problems in n Dimensions

Motivation. For multiple dimensions, the computation of the total particle volume contributes significantly to the complexity of the numerical problem. In fact, solving more general integrals of the type:

$$I_\mu(t) = \int \mu(\mathbf{h}_\mathrm{C})\,\tilde{n}(\mathbf{h}_\mathrm{C},\,t)\,\mathrm{d}\mathbf{h}_\mathrm{C} \tag{4.15}$$

is of central interest since almost all physical properties can be formulated in that way, given an appropriate function $\mu(\mathbf{h}_\mathrm{C})$. For high dimensional cases in the order of $n \approx 100$, deterministic integration methods are clearly outperformed by Monte Carlo methods which is not that clear for the relatively low dimensional problems with $n < 10$ that are treated in the scope of population balances [Press et al., 2007, Schürer, 2003]. However, the concept of Monte Carlo integration seems to be very flexible in its applications so that a complete numerical scheme can be provided in the remainder of this subsection.

Monte Carlo integration. Monte Carlo integration is based on the random sampling of points $\mathbf{h}_{\mathrm{C},i}$ for which the probability density $\widetilde{w}(\mathbf{h}_\mathrm{C},\,t)$ is known. Here, the probability density is already assumed to be time dependent which is essential for the resulting numerical scheme. In addition, note that the position of the sample points $\mathbf{h}_{\mathrm{C},i}$ will also

[2] The distance from the boundary is $\mathbf{a}_{\mathrm{FV}(k),i}^T\mathbf{h}_\mathrm{C}$ and negative for points $\mathbf{h}_\mathrm{C}$ inside of the cone so that γ gradually changes from 0 at $\mathbf{a}_{\mathrm{FV}(k),i}^T\mathbf{h}_\mathrm{C} = -\delta$ to 1 at $\mathbf{a}_{\mathrm{FV}(k),i}^T\mathbf{h}_\mathrm{C} = 0$.

be time dependent: $\mathbf{h}_{\mathrm{C},i}(t)$. Given a total of n_{sample} points, the integral in equation 4.15 is estimated by [Press et al., 2007, Kroese et al., 2011]:

$$I_\mu(t) \approx \frac{1}{n_{\mathrm{sample}}} \sum_i \mu\left(\mathbf{h}_{\mathrm{C},i}(t)\right) \frac{\tilde{n}\left(\mathbf{h}_{\mathrm{C},i}(t),\, t\right)}{\widetilde{w}(\mathbf{h}_{\mathrm{C},i}(t),\, t)}. \tag{4.16}$$

While this approximation itself is a random variable, its standard deviation can be approximated by [Press et al., 2007, Kroese et al., 2011]:

$$\mathrm{std}\left(I_\mu(t)\right) \approx \sqrt{\frac{1}{n_{\mathrm{sample}}\left(n_{\mathrm{sample}}-1\right)} \sum_i \left(\mu(\mathbf{h}_{\mathrm{C},i}(t)) \frac{\tilde{n}(\mathbf{h}_{\mathrm{C},i}(t),\, t)}{\widetilde{w}(\mathbf{h}_{\mathrm{C},i}(t),\, t)} - I_\mu(t)\right)^2}. \tag{4.17}$$

This empirical standard deviation is commonly used as an error measure for Monte Carlo estimates of the integral. While a rigorous error estimator could be derived, this estimate and the coefficient of variation:

$$\delta I_\mu(t) = \frac{\mathrm{std}\left(I_\mu(t)\right)}{I_\mu(t)} \tag{4.18}$$

as a relative error measure are sufficient for applications in this work.

Numerical scheme. To solve the population balance for seeded growth in n dimensions random points are sampled for $t = 0$ from a probability distribution $\widetilde{w}_0(\mathbf{h}_{\mathrm{C}})$ and obtain their positions $\mathbf{h}_{\mathrm{C},i}(0)$, their particle distribution density $\tilde{n}_i(0) = \tilde{n}_0(\mathbf{h}_{\mathrm{C},i})$ and their probability density $\widetilde{w}_i(0) = \widetilde{w}_0(\mathbf{h}_{\mathrm{C},i})$. Applying the method of characteristics, $\mathbf{h}_{\mathrm{C},i}(t)$ is obtained directly and $\tilde{n}_i(t)$ is obtained according to equations 4.1 and 4.9. Since probability is a conservative property of the sampled points and since there is no influx, outflux or net generation in any closed region, it follows:

$$\frac{\partial \widetilde{w}(\mathbf{h}_{\mathrm{C}},\, t)}{\partial t} = -\nabla_{\mathbf{h}_{\mathrm{C}}} \bullet \left(\tilde{\mathbf{g}}\, \widetilde{w}\right) \tag{4.19}$$

analogously to the derivation of the population balance itself [Randolph and Larson, 1971]. The method of characteristics then yields:

$$\frac{\mathrm{d}\widetilde{w}_i}{\mathrm{d}t} = -\widetilde{w}_i \left(\nabla_{\mathbf{h}_{\mathrm{C}}} \bullet \tilde{\mathbf{g}}\right). \tag{4.20}$$

for the time evolution of the probability density. The divergence $(\nabla_{\mathbf{h}_{\mathrm{C}}} \bullet \tilde{\mathbf{g}})$ of the growth field can be estimated numerically so that a closed system of equations is obtained that is comprised of $3 \times n_{\mathrm{sample}}$ ordinary differential equations for $\mathbf{h}_{\mathrm{C},i}(t)$, $\tilde{n}_i(t)$ and $\widetilde{w}_i(t)$ (equations 4.1, 4.9 and 4.20) as well as the integral estimator from equation 4.16 for the total particle volume.

Reduced numerical scheme. Typical problems in this work do not use any source or sink term σ_i so that the particle distribution densities $\tilde{n}_i$ evolve proportional to the respective probability densities $\widetilde{w}_i$. This results in $\tilde{n}_i(t)/\widetilde{w}_i(t)$ being independent of t for each sample point $\mathbf{h}_{\mathrm{C},i}$. Furthermore, the integral estimation in equation 4.16 does only require these fractions. Hence, a simplified numerical scheme is obtained by storing only $\mathbf{h}_{\mathrm{C}}(t)$ which evolves dynamically and:

$$N_i = \frac{1}{n_{\mathrm{sample}}} \frac{\tilde{n}_i(0)}{\widetilde{w}_i(0)} \tag{4.21}$$

which is a constant for each sample point. The estimation of the divergence $(\nabla_{\mathbf{h}_\mathrm{C}} \bullet \tilde{\mathbf{g}})$ is not required and the resulting scheme is comprised of n_sample ordinary differential equations (equation 4.1) and the simplified integral estimator from equation 4.16:

$$I_\mu(t) \approx \sum_i \mu\left(\mathbf{h}_{\mathrm{C},i}(t)\right) N_i. \tag{4.22}$$

Selecting $\widetilde{w}_0(\mathbf{h}_\mathrm{C})$. Importance sampling is an efficient way to improve the accuracy of Monte Carlo estimates and is based on a careful selection of the probability distribution $\widetilde{w}(\mathbf{h}_\mathrm{C},\, t)$. Given the standard deviation of the integral estimate in equation 4.17, the error becomes zero for $\widetilde{w}(\mathbf{h}_\mathrm{C},\, t) = \mu(\mathbf{h}_\mathrm{C}) \frac{\tilde{n}(\mathbf{h}_\mathrm{C},\, t)}{I_\mu(t)}$. This approach, however, is pointless as it requires to know I_μ which is to be evaluated by the selection of $\widetilde{w}(\mathbf{h}_\mathrm{C},\, t)$. On the other hand, any selection of the probability density that is close to this optimum improves the accuracy. For the proposed numerical scheme, only $\widetilde{w}_0(\mathbf{h}_\mathrm{C})$ can be selected, but the integral $I_\mathrm{volume}(t)$ needs to be evaluated several times so that the effort to calculate some $I_\mu(0)$ adds up. For crystal growth, a feasible initial guess is to choose $\widetilde{w}_0(\mathbf{h}_\mathrm{C}) = \frac{\tilde{n}_0(\mathbf{h}_\mathrm{C})}{I_1(0)}$ proportional to the number distribution. The integral $I_1(0)$ can be computed by any integration method provided by Hahn [2005] while the random sampling proportional to $\tilde{n}_0(\mathbf{h}_\mathrm{C})$ is possible by the Metropolis-Hastings algorithm [Metropolis et al., 1953, Haario et al., 2001].

4.3 Case Studies and Summary

4.3.1 Constant Growth

Motivation. The numerical scheme from the previous section provides a new option to trim the errors of a process simulation by the choice of the initial pivot positions $\mathbf{h}_{\mathrm{C},i}(t=0)$. This possibility is based on the error estimator, equation 4.18. The main aim of this case study is to evaluate this error estimator and to demonstrate a strategy to reduce the overall error with runs of small-scale simulations to identify a suitable probability distribution $\widetilde{w}_0(\mathbf{h}_\mathrm{C})$ before a large-scale simulation is started. On a secondary level, the numerical scheme is tested for a rather high-dimensional example with $n_\mathrm{C}=7$. The construction of the initial condition involves the truncation of an otherwise smooth initial condition by the validity cone which might be a typical situation for process simulations. For that reason, only a semi-analytical solution is available as a basis for comparison.

Setup. Modeled is a hypothetic paracetamol crystal that is comprised of the crystal forms $\{1,1,0\}$, $\{0,1,1\}$, $\{0,0,1\}$, $\{2,0,-1\}$, $\{1,0,1\}$, $\{1,0,0\}$ and $\{0,1,0\}$ for which the required decomposition data is computed according to chapter 3. The initial condition is based on a multivariate normal distribution with the mean at $h_{\mathrm{C},i}=10^{-6}\,\mathrm{m}$ and a covariance matrix of $1.5\times10^{-14}\times\mathbf{I}_7\,\mathrm{m}^2$. It is multiplied by 10^{10} which represents the total particle number in a non-truncated case while it must be truncated for the following two reasons. First, it contains crystals that are outside of the validity cone so that the distribution density is set to 0 outside of this cone. Secondly, numerical integration of the distribution is almost impossible when the integration region is chosen too large. For that reason the particle distribution is set to 0 outside a cubic box centered at the mean and extending $3\times10^{-7}\,\mathrm{m}$ in each direction which is 2.4 times the chosen standard deviation. The latter truncated part represents about 7.22×10^9 particles while with both truncations applied, 4.12×10^9 particles are described. The given crystals grow with a constant growth rate:

$$g_{\mathrm{C},i}=10^{-8}\frac{\mathrm{m}}{\mathrm{s}}. \tag{4.23}$$

Basis of comparison. The results are compared against semi-analytical results. Given an arbitrary initial condition and constant growth, an analytic solution of the distribution density is obtained by the method of characteristics for which the characteristic trajectories $\mathbf{h}_\mathrm{C}(t)$ evolve with:

$$\mathbf{h}_\mathrm{C}(t)=\mathbf{h}_\mathrm{C}(0)+\mathbf{g}_{\mathrm{C},i}\,t \tag{4.24}$$

and the number density on these characteristics evolve with:

$$\tilde{n}(\mathbf{h}_\mathrm{C}(t),\,t)=\tilde{n}_0(\mathbf{h}_\mathrm{C}(t)-\mathbf{g}_{\mathrm{C},i}\,t), \tag{4.25}$$

assuming no sink or source terms are present ($\sigma_i=0$). As the validity cone truncates the initial condition, an analytic solution for integrals of the distribution is difficult to obtain so that they are estimated numerically by the VEGAS algorithm, a Monte Carlo method implemented by Hahn [2005].

Results for small-scale simulations. In the first set of results, 100 sample points are generated from various initial probability distributions that are constructed from the initial condition of the problem. Each of these different cases is repeated 10 times with stochastically independent initial sample points $\mathbf{h}_{\mathrm{C},i}$. The first probability distribution uses the unscaled and non-truncated normal distribution that is used for the initial condition. The second case samples the initial points proportional to the number density $\tilde{n}_0$ while the third case samples them proportional to $\mu^{\mathrm{volume}}\tilde{n}$ and the fourth case proportional to a mix of both: $\widetilde{w}_0 \approx \frac{1}{2}\frac{\mu^{\mathrm{volume}}\tilde{n}_0}{I_{\mathrm{volume}}} + \frac{1}{2}\frac{\tilde{n}_0}{I_1}$.

Figure 4.2 shows the evolution of the total particle volume for the 10 runs of the first case as dashed lines and the semi-analytical result as a bold line. Figure 4.3 displays for all four cases the estimated coefficient of variation as dashed lines and the absolute relative error for each run compared against the semi-analytical solution as light blue lines.

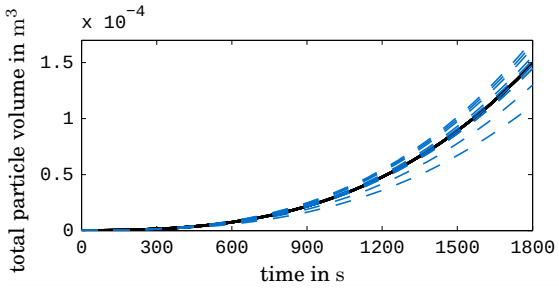

Figure 4.2.: Total particle volume over time by the semi-analytical solution (bold line) and the runs of the numerical scheme for case one (dashed lines)

Discussion for case one. In case one, the initial points $\mathbf{h}_{\mathrm{C},i}$ are sampled from a plain normal distribution. While the overall increase of the total particle volume shown in figure 4.2 is well captured, the relative errors are in the range of 10%. These large errors are caused by the initial sampling for which 59% of the points reside outside of the truncated region of the particle distribution. These points do not bear any information, but they can also not simply be disregarded once the probability distribution $\widetilde{w}_0$ is chosen. In consequence, they result in large terms $(0 - I_{\mathrm{volume}})^2$ for the sum in equation 4.17 and the standard deviation of I_{volume}. Disregarding points with $\tilde{n}_0(\mathbf{h}_{\mathrm{C},i}) = 0$ follows the idea of sampling points proportional to the initial particle distribution $\tilde{n}_0$ which is covered by case two.

Error esimtates. The true coefficient of variation lies within a confidence interval of 88% to 116% of the displayed values in figure 4.3, given a confidence level of 95% [Beyer et al., 1995]. This explains well the spread of the coefficient of variation for the separate runs. Furthermore, assuming I_{volume} being normally distributed, the spread of the relative error is also in good agreement with the coefficient of variation. Overall, it can be concluded that the estimated coefficient of variation is a suitable error measure that can be used to tune the error by the selection of a suitable probability distribution w_0. This is demonstrated by the subsequent three cases while only a single run of the small-scale simulations would be required.

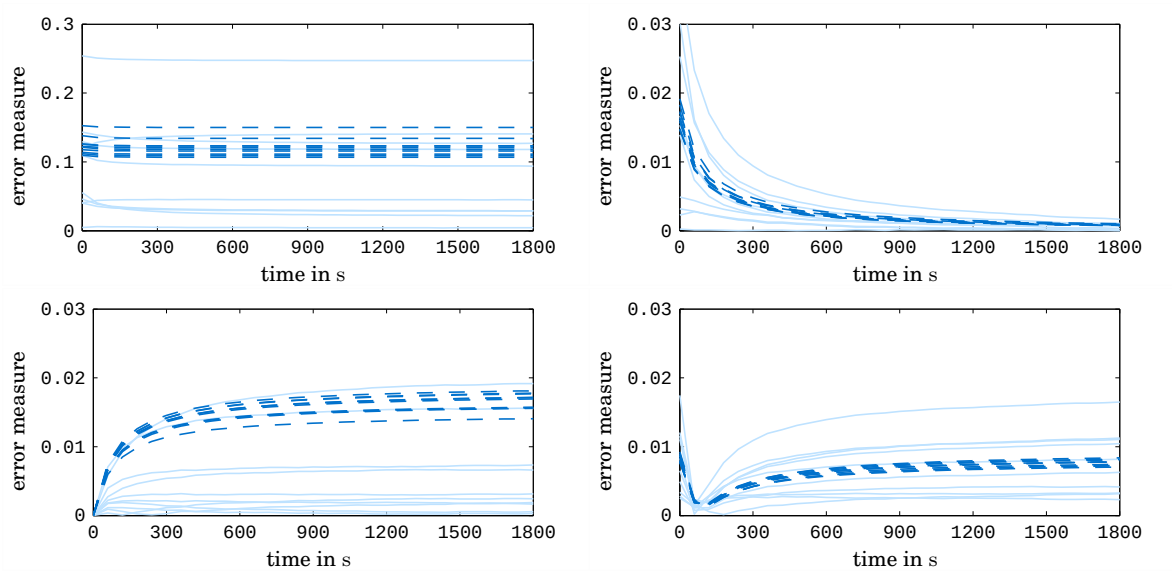

Figure 4.3.: Coefficient of variation (dashed lines) and relative error (light blue lines) for case one (top left, normal distribution), case two (top right, proportional to $\tilde{n}$) case three (bottom left, proportional to $\mu^{\text{volume}}\tilde{n}$) and case four (bottom right, proportional to a mix of $\tilde{n}$ and $\mu^{\text{volume}}\tilde{n}$)

Discussion for case two. For case two, the intuitive approach is followed in which pivots are sampled densely in regions where a lot of particles are present. This implies to choose $\widetilde{w}_0$ proportional to $\tilde{n}_0$. The obtained coefficient of variation (figure 4.3) is below 2% and, hence, significantly better than for case one since, now, all points bear information of the particle distribution. However, comparing the errors for $t = 0$ and $t = 1800\,\text{s}$, the initial particle distribution is not covered well for the computation of the total particle volume. For $t = 0$ the variation of the integrand $\mu^{\text{volume}}\tilde{n}$ is dominated by the variations in μ^{volume} while the variations in μ^{volume} relative to $\tilde{n}$ vanish over time.

Discussion for case three. To increase the accuracy for $t = 0$, the probability density $\widetilde{w}_0$ can be chosen proportional to $\mu^{\text{volume}}\tilde{n}$ which is demonstrated with case three. Following equation 4.17, the error for $t = 0$ is 0. However, the Metropolis sampling applies $\widetilde{w}_0 = \mu^{\text{volume}}\tilde{n}_0/I_{\text{volume}}$ for which I_{volume} is only computed numerical up to a certain accuracy. This uncertainty is not covered by equation 4.17 but can be neglected in this case since I_{volume} is computed with a relative error below 10^{-4}. For the time evolution of the error, it can be concluded that it behaves exactly inverse compared to case two and an analogue discussion is skipped.

Discussion for case four. Since cases two and three behave exactly inverse, they can be mixed to decrease the maximum error that is obtained for all times of the simulation. Therefore, the points are sampled proportional to $\frac{1}{2}\tilde{n}_0/I_1 + \frac{1}{2}\mu^{\text{volume}}\tilde{n}_0/I_{\text{volume}}$ and a coefficient of variation is obtained that is below 1% for all times $0\,\text{s} \leq t \leq 1800\,\text{s}$.

Large-scale simulation. With the tuning of the probability distribution $\widetilde{w}_0$, a large-scale simulation can now be presented that uses 10^4 sample points. The results are shown in figure 4.4 analogous to figure 4.3. Since the numerical integration for the

semi-analytical result used a relative tolerance of 10^{-3} while the coefficient of variation is below 3×10^{-4}, the displayed relative error is dominated by the error of the semi-analytical result.

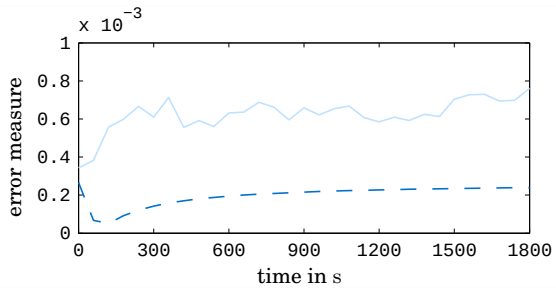

Figure 4.4.: Coefficient of variation (dashed line) and relative error (light blue line) for a large-scale simulation.

Summary. For this case study, a few small-scale simulations are performed with 100 sample points to construct a suitable probability distribution $\widetilde{w}_0$ before a large-scale simulation with 10^4 sample points is executed. This strategy is possible, based on a reliable error estimator that is provided by equation 4.17. A reference solution is not required for this approach even though the tuning for errors might be more complicated for other cases. First, an error in the total particle volume influences the growth rate and subsequently the evolution of the whole system in all time steps thereafter. Secondly, the target measure for error reduction must be carefully chosen and might be supersaturation or a growth rate of some sort instead of the total particle volume. Despite of the utilization of the error estimator, this case study demonstrates the application of the reduced numerical scheme to a complex crystal geometry with a rather high number of facet groups ($n_\mathrm{C} = 7$).

4.3.2 Linear Growth

Motivation. While the case study of the previous subsection focused on the error estimator and a high dimensional process model, this case study focuses on the overall error of the numerical scheme. For this purpose, the distribution densities and four integrals of the distribution are compared against an analytical solution of the problem.

Setup. Modeled are cuboids that use the following matrices for the facet normals and facet grouping:

$$\mathbf{A} = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}^T \qquad \mathbf{M}_{\mathbf{h}_\mathrm{C} \to \mathbf{h}} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}^T . \tag{4.26}$$

The initial condition is a multivariate exponential distribution:

$$\tilde{n}_0(\mathbf{h}_\mathrm{C}) \;=\; \begin{cases} \gamma_\mathrm{N} \left(\frac{1}{\gamma_\mathrm{S}}\right)^{n_\mathrm{C}} \prod_i e^{-\frac{1}{\gamma_\mathrm{S}} h_{\mathrm{C},i}} & \text{for } \mathbf{h}_\mathrm{C} \geq \mathbf{0} \\ 0 & \text{otherwise} \end{cases} \tag{4.27}$$

that is parameterized such that $\gamma_\mathrm{N} = 10^{10}$ equals the total particle number and $\gamma_\mathrm{S} = 10^{-5}\mathrm{m}$ is the mean facet distance for all directions. These given crystals grow with a

linear growth rate:

$$\mathbf{g}(\mathbf{h}_\mathrm{C}) = \mathbf{h}_\mathrm{C} \cdot \frac{1}{\mathrm{s}}. \tag{4.28}$$

Analytic solution. The analytic solution is obtained by the method of characteristics for which the characteristic trajectories $\mathbf{h}_\mathrm{C}(t)$ evolve with:

$$\mathbf{h}_\mathrm{C}(t) = \mathbf{h}_\mathrm{C}(0)\, e^t \tag{4.29}$$

and the number density on these characteristics evolves with:

$$\tilde{n}(\mathbf{h}_\mathrm{C}(t),\, t) = \tilde{n}_0(\mathbf{h}_\mathrm{C}(t)\, e^{-t})\, e^{-n_\mathrm{C} t}, \tag{4.30}$$

assuming no sink or source terms are present ($\sigma_i = 0$).

The total volume of the distribution is given by:

$$I_\mathrm{volume}(t) = 8 \int \tilde{n}(\mathbf{h}_\mathrm{C},\, t)\, h_{\mathrm{C},1} h_{\mathrm{C},2} h_{\mathrm{C},3}\, \mathrm{d}\mathbf{h}_\mathrm{C}.$$

Applying equation 4.30 together with the initial condition, equation 4.27, and performing an integral substitution with $\mathbf{h}_\mathrm{C} = e^t\, \mathbf{h}'_\mathrm{C}$ gives:

$$I_\mathrm{volume}(t) \;=\; \gamma_\mathrm{N} \left(\frac{2}{\gamma_\mathrm{S}}\right)^3 e^{3t} \int_{\mathbf{0}}^{\infty} \prod_i e^{-\frac{1}{\gamma_\mathrm{S}} h'_{\mathrm{C},i}} h'_{\mathrm{C},i}\, \mathrm{d}\mathbf{h}'_\mathrm{C}. \tag{4.31}$$

This integral can be evaluated component-wise to obtain:

$$I_\mathrm{volume}(t) = \gamma_\mathrm{N} \left(2\gamma_\mathrm{S} e^t\right)^3. \tag{4.32}$$

The analytic results for the surface area and mean width follow analogously:

$$I_\mathrm{surface}(t) \;=\; 6\, \gamma_\mathrm{N} \left(2\gamma_\mathrm{S} e^{\gamma_\mathrm{G} t}\right)^2 \tag{4.33}$$

$$I_\mathrm{mean\,width}(t) \;=\; 3\, \gamma_\mathrm{N}\, \gamma_\mathrm{S}\, e^{\gamma_\mathrm{G} t} \tag{4.34}$$

while the total particle number is constant:

$$I_1(t) = \gamma_\mathrm{N}. \tag{4.35}$$

Numerical setup. A total of 10^6 points are sampled initially from the probability distribution $\widetilde{w}_0 = \frac{1}{2}\tilde{n}/I_1 + \frac{1}{2}\mu^\mathrm{volume}\tilde{n}/I_\mathrm{volume}$ where $I_1 = \gamma_\mathrm{N}$ is used and I_volume is computed at $t = 0$ by equation 4.32. The Markov chain on which the Metropolis algorithm is based evaluated $\widetilde{w}_0$ about 1.7×10^8 times so that the sampling of these points takes about 42 minutes [Desktop Computer, 2009]. The system is integrated for $t \in [0,\, 3\,\mathrm{s}]$ where a snapshot is created every $0.2\,\mathrm{s}$. While the integration of the numerical scheme takes only 16 minutes, the accuracy of the results is limited by the required memory space of $1.1\,\mathrm{Gb}$ to store the result. This memory limitation does not allow a significant increase of the total number of sample points to improve the presented accuracy.

Results. The time evolution of the total particle number, volume, surface area and mean width is shown in figure 4.5 based on equations 4.32 to 4.35. The total mean width increases from $3 \times 10^5\,\mathrm{m}$ to $6 \times 10^6\,\mathrm{m}$ and implies a significant increase of the modeled particle sizes. The maximum of the absolute relative errors for the distribution densities over time is shown in figure 4.6. While this error constantly increases, the order of the error originates from the time integrator for the system of ordinary differential equations. This assumption was successfully tested by lowering the relative tolerance of the time integrator. Table 4.1 lists the coefficient of variation and the obtained relative errors. All relative errors are below 5×10^{-4} while the coefficient of variation suggests a tolerance of roughly 1.7×10^{-3}, given a confidence level of 95% and assuming the integrals being normally distributed.

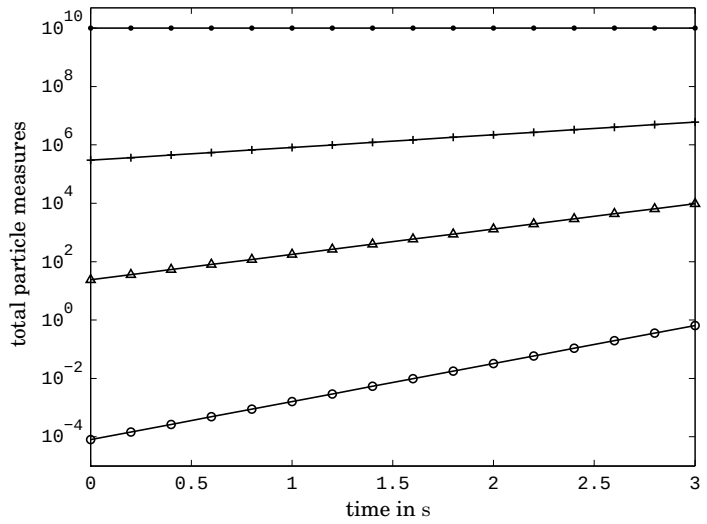

Figure 4.5.: Time evolution of the total particle number (marker dots), mean width in m (pluses), surface area in m^2 (triangles) and volume in m^3 (circles).

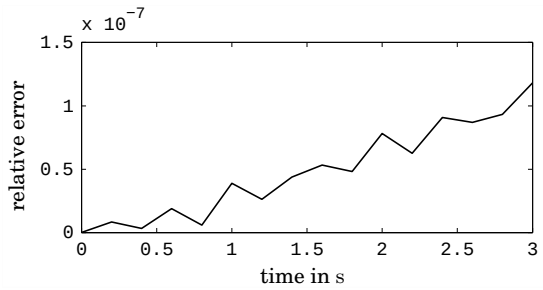

Figure 4.6.: Time evolution of the maximum absolute error for $\tilde{n}(\mathbf{h}_{\mathrm{C},i}(t),\, t)$.

Summary. This case study successfully validates the proposed numerical scheme. While it demonstrates a high level of accuracy, it also shows that the accuracy of the proposed scheme is limited by the total number of sample points that can be stored. As

Table 4.1.: Error measures for total values of the particle distribution.

	number	volume	surface area	mean width
coefficient of variation	6.69×10^{-4}	6.69×10^{-4}	5.36×10^{-4}	5.28×10^{-4}
relative error	2.20×10^{-4}	2.20×10^{-4}	2.11×10^{-4}	4.96×10^{-4}

the error decays only with $1/\sqrt{n_{\text{sample}}}$, the selection of the initial probability distribution is essential to obtain a high accuracy with this proposed scheme.

5

Modeling of Rounded Shapes for Attrition or Dissolution

5.1 Introduction and Preliminaries

5.1.1 Introduction

Focus. This section introduces a shape model for convex rounded particles. In the derivations and applications, it focuses specifically on the attrition of crystals while rounded particles also occur frequently in geology as sand grains, boulders or rocks or are formed by the dissolution of crystals. Attrition is caused by crystals colliding in an agitated vessel either with themselves or with the stirrer or walls. If the collision energy is not large enough to break the particle, only small fragments are separated and the shape is altered slightly towards a smaller crystal with rounded edges and vertices. Attrition is important for crystallization processes since it is almost always present and has a large influence on the obtained size distribution. The generated particle fragments provide nuclei that grow and, therefore, deplete the supersaturation in the system. The generation of small fragments and their growth, results in a size distribution that is broadened and shifted in the direction of smaller particles, compared to a process without attrition. While this suggests that primarily the number and size of the separated particles are influencing the process, it must be considered that these fragments are typically very small and difficult to measure. Measuring the shape change of the original particles might therefore provide an alternative way to track the generation of such nuclei. Additionally, rounded particles are typically less affected by collisions than angular particles which might be an important factor for predictive modeling.

Determining attrition rates. Attrition kinetics are often determined based on a power law assumption and experimental data [Meadhra et al., 1996, Hounslow et al., 2005]. However, these estimated rates are typically not predictive and applicable to significantly different systems. These differences particularly include the size of the system which makes it difficult to predict the behavior of the final industrial process from laboratory experiments [Westhoff and Kramer, 2012]. In contrast to this approach, mechanistic models consider the local geometry at the impact site and include basic material properties [Gahn and Mersmann, 1997, Briesen, 2007, 2008]. The considered geometrical detail comprises the opening angle of a cone whose tip represents the im-

pact site. While these models are designed to result in a better predictability, they must be augmented by models that predict the frequency of collisions with respect to the impact angle and the impact velocity. Since industrial systems often contain a high particle mass fraction, obtaining such rates is difficult [Trzeciak et al., 2004, Heine and Pratsinis, 2007, Reinhold, 2008, Reinhold and Briesen, 2012]. Even more important, the mechanistic models above state the attrition rate being dependent on the overall particle shape while the work by Briesen [2009a] seems to be the only one that actually considers the corresponding shape changes for attrition rates. His shape model assumes a cube for which the rounding is modeled by chipping off its vertices to obtain the facets of an octahedron like it is shown in figure 5.1. Apparently, this shape model cannot reflect the rounding of arbitrary crystal geometries since it is based on a cube and it does only present a crude rounding by the application of the octahedron facets. In summary of the mentioned attrition modeling approaches, an improved shape model of rounded particles closes one of the gaps to obtain predictive attrition kinetics. However, the detailed analysis of attrition mechanics is clearly beyond the scope of this work.

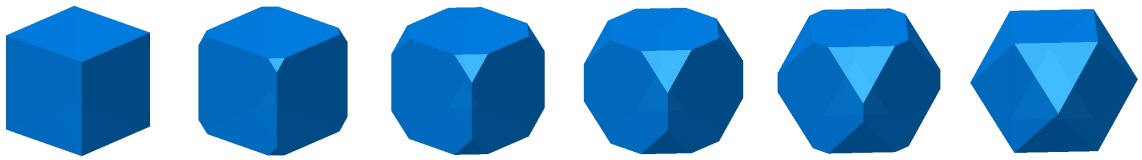

Figure 5.1.: Sample shapes for the attrition (or roundness) model by Briesen [2009a]: fully angular on the left and fully attrited (or fully roundned) on the right

Measuring roundness. Typical measurements for a crystallization system consider the one-dimensional size distribution and the bulk solute concentration. While several imaging techniques in the context of crystallization processes exist [Schorsch et al., 2012, Singh et al., 2012], none of them considers the rounded shapes that occur during attrition. On the other hand, roundness is commonly measured in mineralogy, geology or related fields. There, roundness or angularity is interpreted as an independent property from the overall shape or texture [Barret, 1980]. While shape is a global property of a particle, roundness or angularity are properties of the corners and texture is a property in even smaller size scales. Numerous definitions for roundness exist that do not follow this separation and contain overall shape information [Pons et al., 1999, Masad, 2000, Al-Rousan et al., 2007, Hentschel and Page, 2003]. In contrast, the definitions by Wadell [1932] or Powers [1953] clearly separate roundness from the overall shape. Powers [1953] proposed to classify the roundness of particles into 6 categories from 'very angular' to 'well rounded' while the classification is based on human observation skills. It is, hence, not applicable for the automatic monitoring of crystallization processes. Wadell [1932] provides a definition of roundness that is based on a specific measurement procedure. Three perpendicular projections are taken, like the three images on the left of figure 5.2. For each projection k, the radius R_k of a maximum inscribed circle (bold circles) and the curvature radii $r_{k,i}$ for all corners (thin circles) are measured. Corners are defined by regions of the boundary for which the curvature is less than or

equal to R_k. The final roundness is then obtained by:

$$\text{roundness} = \frac{1}{N} \sum_{i,k} \frac{r_{k,i}}{R_k} \tag{5.1}$$

where N is the total number of measured curvature radii. An outstanding property of this definition is that a roundness of 1 for a single projection implies a fully rounded shape so that the same projection cannot become any more rounded. The outer right example in figure 5.2 demonstrates such a situation. However, this only holds for an single isolated projection. All four images in figure 5.2 are projections of the same particle which demonstrates that the procedure by Wadell not necessarily result in a roundness of 1 for fully rounded particles. While this procedure was also designed to rely on human observation skills, Maerz [2004] proposed an image analysis procedure that evaluates the radii of curvature automatically. Since his work focuses on angularity, the measurement identifies only a local minimum for the curvature radii. Another measurement procedure by Lee et al. [2007] uses three dimensional laser scanning data and computes the roundness based on morphological image analysis. However, today no technical equipment exists to perform similar measurements during a crystallization process. In summary, roundness measurements of particles during crystallization are not yet available while numerous different definitions of roundness or angularity exist in the field of mineralogy or geology. Even though the corresponding measurement procedures are not suitable for crystallization processes and do not fit well to the new shape model, the various definitions point out desirable properties of a roundness quantification that are utilized in the next subsection.

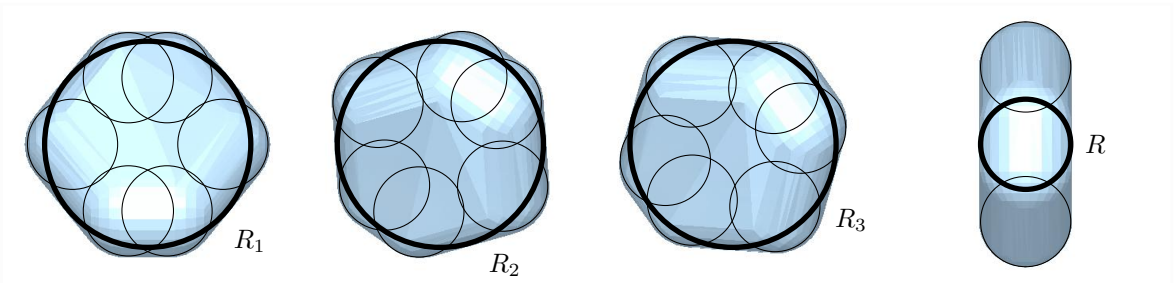

Figure 5.2.: Illustration of roundness by Wadell [1932]

Outlook. The following subsection introduces the new shape model. It is based on the Minkowski addition of a polytope and a ball so that it can reproduce perfectly rounded edges and vertices like it is shown in figure 5.3. Section 5.2 then consolidates the new shape model with the framework from chapter 3. It additionally analyzes how the dynamics of rounding by attrition or the de-rounding by growth are properly described and isolates required parameters. Section 5.3 finally presents a shape identification procedure to measure particle size and roundness in a crystallization process. The procedure is tested for the proof of principle based on simulated particles and based on experimental data.

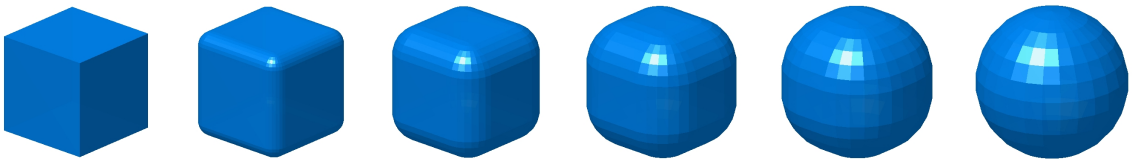

Figure 5.3.: Sample shapes for the roundness model in this work: full angularity on the left and full roundness on the right (the visible facets at the rounded edges and vertices are rendering artifacts)

5.1.2 Shape Model

Principles. Rounded edges and vertices of an otherwise faceted crystal are well represented by the Minkowski sum of a ball (a solid sphere), representing the rounding, and a polyhedron, representing the facets. An example of such a Minkowski sum is shown in figure 5.4 where the summands are drawn in the left and the sum is drawn in the right. Note the facets of the polytope summand also being present in the sum (illustrated by the coloring of the sum) while their distances have increased. The polytope summand is called the kernel crystal or kernel polytope $P(\mathbf{h}_\mathrm{k})$ and given in $\mathcal{H}$-representation. Together with the unit ball B that is scaled to the radius λ_r, the rounded shape is defined by:

$$C(\mathbf{h}_\mathrm{k},\, \lambda_\mathrm{r}) = P(\mathbf{h}_\mathrm{k}) + \lambda_\mathrm{r} B. \tag{5.2}$$

The support planes of the polytope summand $P(\mathbf{h}_\mathrm{k})$ are illustrated for the rounded shape $C(\mathbf{h}_\mathrm{k},\, \lambda_\mathrm{r})$ by the black frame in the outer right of figure 5.4. They form a polytope that circumscribes the sum so that the rounded shape $C(\mathbf{h}_\mathrm{k},\, \lambda_\mathrm{r})$ could have evolved by abrasion from this black framed polyhedron. Because of this relation, the circumscribing polytope $P(\mathbf{h}_\mathrm{k} + \lambda_\mathrm{r})$ is valuable and its facet distances are abbreviated by:

$$\mathbf{h}_\mathrm{c} = \mathbf{h}_\mathrm{k} + \lambda_\mathrm{r}. \tag{5.3}$$

In mathematical literature, the bodies that are created by equation 5.2 are known as parallel bodies. They are well studied and the so-called Steiner formula provides equations to determine the volume, surface area and mean width of parallel bodies based on the same properties with respect to the summands [Santaló, 1976, Schneider, 2008, Hadwiger, 1957]:

$$\begin{aligned} \mu^{\text{volume}}\left(C(\mathbf{h}_\mathrm{k},\, \lambda_\mathrm{r})\right) &= \mu^{\text{volume}}\left(\mathbf{h}_\mathrm{k}\right) + \mu^{\text{surface}}\left(\mathbf{h}_\mathrm{k}\right)\, \lambda_\mathrm{r} \\ &\quad + 2\pi\, \mu^{\text{mean width}}\left(\mathbf{h}_\mathrm{k}\right)\, \lambda_\mathrm{r}^2 + \frac{4\pi}{3}\, \lambda_\mathrm{r}^3 \end{aligned} \tag{5.4}$$

$$\mu^{\text{surface}}\left(C(\mathbf{h}_\mathrm{k},\, \lambda_\mathrm{r})\right) = \mu^{\text{surface}}\left(\mathbf{h}_\mathrm{k}\right) + 4\pi\, \mu^{\text{mean width}}\left(\mathbf{h}_\mathrm{k}\right) \lambda_\mathrm{r} + 4\pi\, \lambda_\mathrm{r}^2 \tag{5.5}$$

$$\mu^{\text{mean width}}\left(C(\mathbf{h}_\mathrm{k},\, \lambda_\mathrm{r})\right) = \mu^{\text{mean width}}\left(\mathbf{h}_\mathrm{k}\right) + 2\, \lambda_\mathrm{r}. \tag{5.6}$$

Comparison to prior work. Briesen [2009a] already provided a shape model for attrition. The initial non-rounded shapes are cubes with facet distances of length L_1 (see figure 5.1 on the outer left). The rounding effect is described by the facets of an octahedron with facet distances given by the shape parameter L_2. The fully rounded

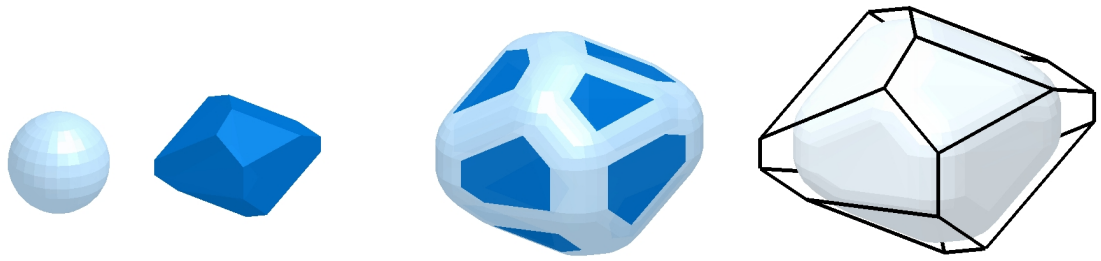

Figure 5.4.: Decomposition of the shape model with the ball $\lambda_r B$ (light blue, outer left), the kernel polytope $P(\mathbf{h}_k)$ (blue, left), the rounded shape $C(\mathbf{h}, \lambda_r)$ (right) and the circumscribing polytope $P(\mathbf{h}_c)$ (black frame, outer right)

shape is then obtained by the cube-octahedron in the outer right of figure 5.1 that uses $L_2 = \frac{2}{\sqrt{3}} L_1$. This shape model can be rewritten analogously to equation 5.2 with a kernel cube $P(h_{\text{cube}})$ that uses new facet distances h_{cube} and the cube-octahedron shape $S_{\text{cube-octahedron}}$ that uses $L_1 = 1$ and is scaled by the parameter λ:

$$C'(h_{\text{cube}}, \lambda) = P(h_{\text{cube}}) + \lambda S_{\text{cube-octahedron}}. \tag{5.7}$$

It applies the coordinate transformation:

$$\begin{pmatrix} L_1 \\ L_2 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ \sqrt{3} & \frac{2}{\sqrt{3}} \end{pmatrix} \begin{pmatrix} h_{\text{cube}} \\ \lambda \end{pmatrix}. \tag{5.8}$$

Therefore, the shape model by equation 5.2 represents an enhancement and generalization of Briesen's work. It allows to describe the rounding of arbitrary polytopes with perfectly rounded edges.

Quantifying roundness. The numerous definitions of roundness, discussed in the introduction, are not suitable for the new shape model. Most of them do not isolate roundness from an overall shape description while others are based on human observation skills or contain variables that are not computed easily by the shape model. However, they provide a general guideline for roundness descriptors. First, roundness should be a property that is independent from the general shape or texture. Secondly, it should be size-independent so that its quantification includes the scaling by some size property. Thirdly, roundness is a property with two boundaries: one for fully angular shapes and one for fully rounded shapes. Choosing a range from 0 for fully angular shapes to 1 for fully rounded shapes is, hence, a suitable choice.

The chosen quantification of roundness in this work follows these guidelines and reflects the intrinsic nature of the new shape model. Given an arbitrary size measure, the difference of the total size and the size of the kernel polytope results in that part of the total size that is generated by the Minkowski addition of the ball. The roundness descriptor is based on this relation and, therefore, called additive roundness. It is given

based on the mean width, surface area and volume:

$$\mu^{B1} = 1 - \frac{\mu^{\text{mean width}}(\mathbf{h}_\text{k})}{\mu^{\text{mean width}}(C(\mathbf{h}_\text{k}, \lambda_r))} = \frac{2\lambda_\text{r}}{\mu^{\text{mean width}}(\mathbf{h}_\text{k}) + 2\lambda_\text{r}} \tag{5.9}$$

$$\mu^{B2} = \left(1 - \frac{\mu^{\text{surface}}(\mathbf{h}_\text{k})}{\mu^{\text{surface}}(C(\mathbf{h}_\text{k}, \lambda_r))}\right)^2 \tag{5.10}$$

$$\mu^{B3} = \left(1 - \frac{\mu^{\text{volume}}(\mathbf{h}_\text{k})}{\mu^{\text{volume}}(C(\mathbf{h}_\text{k}, \lambda_r))}\right)^3. \tag{5.11}$$

The symbol μ^{Bn} indicates the unit ball B that is used in the shape model and the n^{th} power which is used according to the dimensionality of the measure to adjust the realized values between 0 and 1 to the perceived roundness of the examples in figure 5.5. Not considering this scaling results only in minor variations of the shape for μ^{B3} being between 0 and 0.5 while a large range of roundness remains for the range between 0.9 and 1.

The first row of figure 5.5 applies a mean width based additive roundness of $\mu^{B1} = 0.5$. The kernel polytope in the left case is a line segment so that the surface and volume based additive roundness are $\mu^{B2} = 1$ and $\mu^{B3} = 1$, respectively. These shapes are already fully rounded so that a roundness of 1 is actually the proper interpretation. The second row shows sample shapes with a surface area based additive roundness of $\mu^{B2} = 0.5$. Here, the kernel polytope in the middle is a flat plate and the shape is fully rounded. This situation is only properly reflected by the volume based additive roundness, $\mu^{B3} = 1$. The third row displays sample shapes that use $\mu^{B3} = 0.5$ and all shapes appear to be equally rounded. It can be concluded from these observations that the volume based additive roundness is the only measure that properly assigns a roundness of 1 to fully rounded shapes. However, the total volume is difficult to measure based on typical imaging techniques as they are applied in section 5.3. In these cases, the mean width based additive roundness is a reasonable compromise.

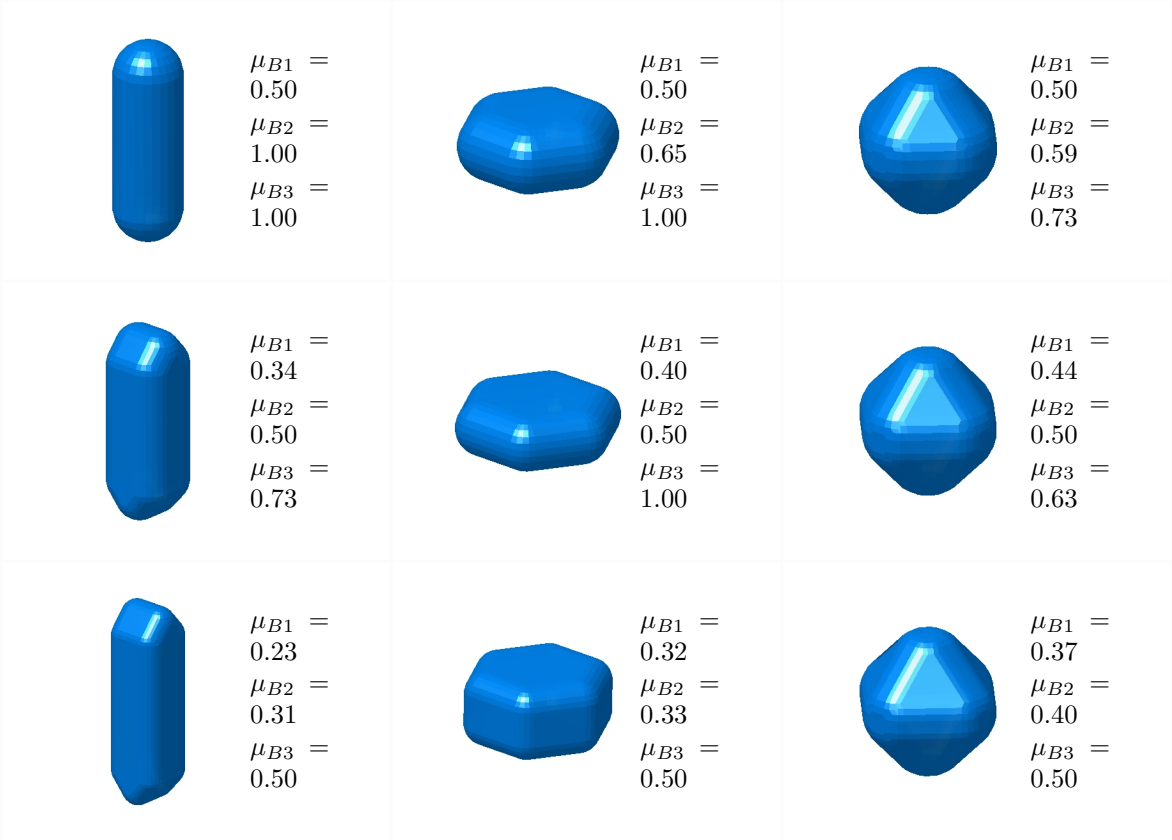

Figure 5.5.: Display of shapes for additive roundness with urea samples (left, kernel polytope in 1st row is a line segment), succinic acid samples (middle column; kernel polytopes in 1st and 2nd row are flat plates) and potash alum samples (right column)

5.2 Modeling Attrition Processes

5.2.1 Geometrical State Space

Motivation. The challenges of shape representations in general consist of possibly inconsistent combinations of shape coordinates, zero-size boundaries and measure computation as it is discussed in the introduction to chapter 3. This does not change for the modeling of rounded particles. However, the new shape model is a consistent extension of the existing framework for $\mathcal{H}_\mathrm{C}$-representations since both approaches are linked to the Minkowski addition of convex bodies. This subsection analyses the geometrical state space and demonstrates that the shape model for rounded particles is conform with the existing framework for proper $\mathcal{H}_\mathrm{C}$-representations.

State space. While the original shape model is based on a $\mathcal{H}$-representation for the kernel polytope, the geometrical state space should be based on a suitable $\mathcal{H}_\mathrm{C}$-representation to reduce the number of shape coordinates. Therefore, the vector $\mathbf{h}_\mathrm{k}$ (n_H-dimensional) is substituted by a new vector $\mathbf{h}_\mathrm{C,k}$ (n_C-dimensional) from the chosen $\mathcal{H}_\mathrm{C}$-representation:

$$C(\mathbf{h}_\mathrm{C,k},\, \lambda_\mathrm{r}) = P(\mathbf{h}_\mathrm{C,k}) + \lambda_\mathrm{r} B. \tag{5.12}$$

The geometrical state space then comprises the vectors:

$$\mathbf{h}_\mathrm{R} = \begin{pmatrix} \mathbf{h}_\mathrm{C,k} \\ \lambda_\mathrm{r} \end{pmatrix} \tag{5.13}$$

where the index R indicates the represented roundness or the appended radius parameter. This shape model describes a ball of radius λ_r for $\mathbf{h}_\mathrm{C,k} = \mathbf{0}$ which results in $P(\mathbf{h}_\mathrm{C,k}) = \{\mathbf{0}\}$. A perfectly angular shape is described by $\lambda_\mathrm{r} = 0$.

General validity. Equation 5.12 results in reasonable shapes when the vector $\mathbf{h}_\mathrm{C,k}$ represents a non-empty polytope and the radius λ_r is not negative. For a complete understanding of the geometrical state space, however, the other cases must also be considered.

First, the case $P(\mathbf{h}_\mathrm{C,k}) = \emptyset$ with $\lambda_\mathrm{r} \geq 0$ is considered for which equation 5.12 computes the shape $C(\mathbf{h}_\mathrm{C,k},\, \lambda_\mathrm{r}) = \emptyset$. The upper left shape from figure 5.5 is a valuable example to demonstrate a principle problem of this interpretation. There, the kernel polytope is a line segment so that the vector $\mathbf{h}_\mathrm{C,k}$ resides on the boundary of the non-empty polytope cone $\mathcal{C}_\mathrm{FV(0)}$ of the chosen $\mathcal{H}_\mathrm{C}$-representation. Shifting the vector $\mathbf{h}_\mathrm{C,k}$ only slightly outside of the non-empty polytope cone results in an empty kernel polytope and the whole particle disappears at once. This is not reasonable. A vector $\mathbf{h}_\mathrm{C,k}$ with $P(\mathbf{h}_\mathrm{C,k}) = \emptyset$ is invalid and an idea might be to map the vector $\mathbf{h}_\mathrm{C,k}$ to a valid representative. But such a validity mapping does not exist since a vector $\mathbf{h}_\mathrm{C,k} \notin \mathcal{C}_\mathrm{FV(0)}$ is always mapped to $\mathbf{h}_\mathrm{C,k} = \mathbf{0}$ which results in an immediate jump to a sphere. This is also not reasonable. It follows that an $\mathbf{h}_\mathrm{R}$-vector with $\mathbf{h}_\mathrm{C,k} \notin \mathcal{C}_\mathrm{FV(0)}$ and $\lambda_\mathrm{r} \geq 0$ is generally problematic and it remains to represent an empty shape with $C(\mathbf{h}_\mathrm{C,k},\, \lambda_\mathrm{r}) = \emptyset$. Even though this seems to be a severe drawback, this decision is unproblematic in applications since disappearing kernel polytopes always lead to a volume based additive roundness of $\mu^{B3} = 1$ and the kinetic rates in subsection 5.2.2 ensure that the kernel polytope cannot decrease in size any further.

Secondly, the case $P(\mathbf{h}_{\mathrm{C,k}}) \neq \emptyset$ with $\lambda_\mathrm{r} < 0$ is considered for which equation 5.12 has two valid interpretations. In the first case the negative sign belongs to the scaling factor of the unit ball and implies a point reflection at the origin so that $-\lambda_\mathrm{r} B = \lambda_\mathrm{r} B$ and $C(\mathbf{h}_{\mathrm{C,k}},\, \lambda_\mathrm{r}) = C(\mathbf{h}_{\mathrm{C,k}},\, -\lambda_\mathrm{r})$ holds. In the second interpretation, the Minkowski addition is altered to a Minkowski difference. In contrast to the ordinary scalar addition and subtraction, the Minkowski difference is not the reverse operation of the Minkowski addition. If this interpretation is applied, the resulting shape for a negative radius yields: $C(\mathbf{h}_{\mathrm{C,k}},\, \lambda_\mathrm{r}) = P(\mathbf{h}_{\mathrm{C,k}} - \lambda_\mathrm{r})$ and, hence, can be expressed by a different vector: $\mathbf{h}_\mathrm{R} = \left(\mathbf{h}_{\mathrm{C,k}}^T + \lambda_\mathrm{r},\, 0\right)^T$. Even though these two interpretations are possible, it is more reasonable to treat the case $\mathbf{h}_{\mathrm{C,k}} \in \mathcal{C}_{\mathrm{FV}(0)}$ with $\lambda_\mathrm{r} < 0$ as an invalid vector. The appropriate direction for the validity mapping is discussed in subsection 5.2.3 (equation 5.35) together with an appropriate example. It results in: $C(\mathbf{h}_{\mathrm{C,k}},\, \lambda_\mathrm{r}) = C(\mathbf{h}_{\mathrm{C,k}} + \lambda_\mathrm{r},\, -\lambda_\mathrm{r})$ and, hence, combines both interpretations.

Lastly, when $P(\mathbf{h}_{\mathrm{C,k}}) = \emptyset$ and $\lambda_\mathrm{r} \leq 0$ holds, then it is feasible to assume that the shape $C(\mathbf{h}_{\mathrm{C,k}},\, \lambda_\mathrm{r})$ is also empty. Such a vector can be represented by $\mathbf{h}_\mathrm{R} = \mathbf{0}$ so that the corresponding validity mapping is also consistent with $\mathcal{H}_\mathrm{C}$-representations.

Validity cones. The above discussion suggests that the concept of validity can be handled similarly to $\mathcal{H}_\mathrm{C}$-representations. The corresponding regions are defined below while the new notation is summarized in the subsequent paragraph. A vector $\mathbf{h}_\mathrm{R}$ for equation 5.12 is considered valid when it provides a valid representation of the kernel polytope $P(\mathbf{h}_{\mathrm{C,k}})$ and a positive radius λ_r so that the corresponding region in the $\mathbf{h}_\mathrm{R}$-space is given by:

$$\mathbf{A}_{\mathrm{V/R}} \mathbf{h}_\mathrm{R} \;\leq\; \mathbf{0} \tag{5.14}$$

$$\mathbf{A}_{\mathrm{V/R}} \;=\; \begin{bmatrix} \mathbf{A}_\mathrm{V} & \mathbf{0} \\ \mathbf{0} & -1 \end{bmatrix}. \tag{5.15}$$

Additionally, the validity of separate facets that also indicates the existence of these facets transfers directly to the new shape model because of the properties of the Minkowski addition. The facet validity cone $\mathcal{C}_{\mathrm{FV}(i)/\mathrm{R}}$ for facet i is given by its facet normal matrix:

$$\mathbf{A}_{\mathrm{FV}(i)/\mathrm{R}} = \begin{bmatrix} \mathbf{A}_{\mathrm{FV}(i)} & \mathbf{0} \end{bmatrix}. \tag{5.16}$$

This condition is independent on the ball radius λ_r. A similar cone $\mathcal{C}_{\mathrm{FV}(B)/\mathrm{R}}$ for the validity of the radius λ_r uses the facet normal matrix:

$$\mathbf{A}_{\mathrm{FV}(B)/\mathrm{R}} = \begin{bmatrix} \mathbf{0} & -1 \end{bmatrix}. \tag{5.17}$$

Lastly, the shape $C(\mathbf{h}_{\mathrm{C,k}},\, \lambda_\mathrm{r})$ is considered to be an empty set when the kernel polytope is empty so that the non-empty shape cone (analogue to the non-empty polytope cone) is given by the facet normal matrix according to equation 5.16.

Measures. Measure computation is essentially based on the measure calculation for the vectors $\mathbf{h}_{\mathrm{C,k}}$ according to equations 5.4 to 5.6. Feret diameters, edge lengths or selected surface areas yield similar formulas that can be easily derived. Hence, the concept of measure computation for $\mathcal{H}_\mathrm{C}$-representations transfers seamlessly to the shape model for rounded particles.

Notation. Since the new shape model is a consistent extension of the concepts that are introduced in chapter 3, the corresponding notation transfers likewise. When the subscript contains the letter C for 'constrained', it is exchanged by R for 'rounded'. When the letter C is not contained in a symbol, the subscript is extended by $/\mathrm{R}$, reading 'in $\mathbf{h}_\mathrm{R}$-space'.

5.2.2 Attrition Dynamics

Motivation. While the shape model from equation 5.2 or 5.12 represents rounded shapes, it is not yet clear how an attrition rate can be mapped to the parameters $\mathbf{h}_{\mathrm{C,k}}$ and λ_r and how the rounding or de-rounding effect can be incorporated. Therefore, this subsection derives equations that convert a given increase or decrease of the mean width (the attrition rate) to changes in the shape parameters $\mathbf{h}_{\mathrm{C,k}}$ and λ_r. Since a fully rounded particle cannot become any more rounded, two separate mechanisms will be required. The first mechanism realizes a pure rounding or de-rounding of the particle shape. The second mechanism assumes a fully rounded particle for which roundness is preserved.

Discussion. Figure 5.6 demonstrates the rounding of a shape in a close up view. The original angular shape is highlighted by the boundary of the blue area. It equals the circumscribing polytope that was already introduced in subsection 5.1.2 (equation 5.3) and remains constant. The actual shape $C(\mathbf{h}_\mathrm{k},\, \lambda_\mathrm{r})$ is drawn by a bold black line and becomes increasingly rounded from the left to the right. The illustrated evolution shows a pure rounding of the shape while the transition in the middle demonstrates that the disappearance of facets is contained seamlessly. If the same figure is interpreted from the right to the left, it demonstrates the de-rounding due to growth of the rounded parts of the shape. The middle figure then demonstrated the appearance of new facets. It can be concluded that constraining the distances of the circumscribing polytope to remain constant[1]:

$$\frac{\mathrm{d}\mathbf{h}_\mathrm{c}}{\mathrm{d}t} = \mathbf{0} \tag{5.18}$$

results in a pure rounding or de-rounding.

Derivation. Given the constraint in equation 5.18, the shape model leaves only one degree of freedom that must necessarily be used to represent the increase or decrease in size. In the following, it is assumed that the governing size property is the mean width. The time derivative of the mean width, as it is given in equation 5.6, is:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mu^{\text{mean width}}\left(\mathbf{h}_\mathrm{k}, \lambda_\mathrm{r}\right) = \left.\frac{\partial \mu^{\text{mean width}}\left(\mathbf{h}'_\mathrm{k}\right)}{\partial \mathbf{h}'_\mathrm{k}}\right|_{\mathbf{h}_\mathrm{k}} \cdot \frac{\mathrm{d}\mathbf{h}_\mathrm{k}}{\mathrm{d}t} + 2\frac{\mathrm{d}\lambda_\mathrm{r}}{\mathrm{d}t}. \tag{5.19}$$

Using equations 5.3 and 5.18, it results:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mu^{\text{mean width}}\left(\mathbf{h}_\mathrm{k}, \lambda_\mathrm{r}\right) = \left(2 - \left(\left.\frac{\partial \mu^{\text{mean width}}\left(\mathbf{h}'_\mathrm{k}\right)}{\partial \mathbf{h}'_\mathrm{k}}\right|_{\mathbf{h}_\mathrm{k}} \cdot \mathbf{1}\right)\right)\frac{\mathrm{d}\lambda_\mathrm{r}}{\mathrm{d}t} \tag{5.20}$$

where $\mathbf{1}$ is a vector of ones in the appropriate dimension. Note that the mean width is typically not differentiable at the boundary of a unified partition since the coefficients

[1] The facet distances of the circumscribing polytope $\mathbf{h}_\mathrm{c}$ (equation 5.3) are not to be confused with the facet distances $\mathbf{h}_{\mathrm{C,k}}$ of the kernel polytope.

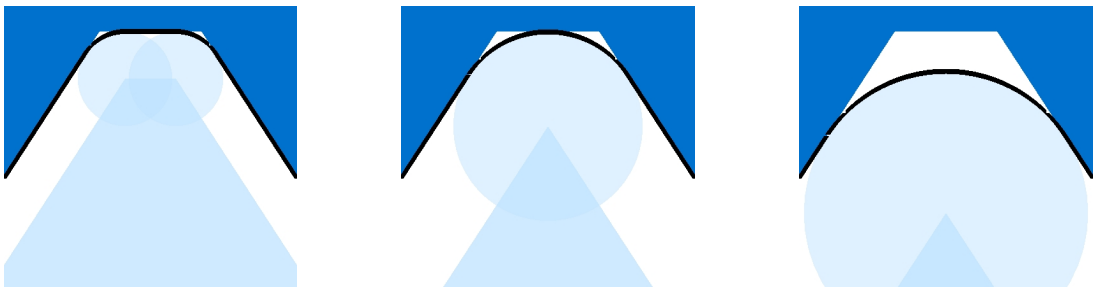

Figure 5.6.: Details of sample shapes to demonstrate the effect of pure rounding (from left to right) or de-rounding (from right to left) while the shape is drawn with a bold black line, the circumscribing polytope is represented by the boundary of the blue area, the angular light blue area highlights the kernel polytope and the light blue circles illustrate the radius λ_r

for measure computation change at these positions in state space. Normal unified partition boundaries, however, are not problematic for dynamic simulations since they are passed in the time integration and a possible numerical error is bounded by the chosen step size in time. However, the situation when the vector $\mathbf{h}'_\mathrm{k}$ approaches the boundary of the validity cone must be handled carefully and equation 5.19 does not apply in these cases. To resolve this issue, the partial derivative is computed numerically based on a directional finite difference:

$$\left.\frac{\partial \mu^{\text{mean width}}\left(\mathbf{h}'_\mathrm{k}\right)}{\partial \mathbf{h}'_\mathrm{k}}\right|_{\mathbf{h}_\mathrm{k}} \cdot \mathbf{1} \approx \frac{\mu^{\text{mean width}}\left(\mathbf{h}_\mathrm{k}\right) - \mu^{\text{mean width}}\left(\mathbf{h}_\mathrm{k} - \frac{\delta}{\sqrt{n_\mathrm{C}}}\mathbf{1}\right)}{\delta} \tag{5.21}$$

using a suitably small value δ. This approach averages the partial derivative in an appropriate manner and neglects singularities. It is similar to the handling of stiffness in subsection 4.2.1. In the final result, equations 5.3 and 5.20 can be used to transfer a physical rate in terms of the mean width: $\frac{\mathrm{d}}{\mathrm{d}t}\mu^{\text{mean width}}\left(\mathbf{h}_\mathrm{k}, \lambda_\mathrm{r}\right)$ to the rates $\frac{\mathrm{d}\mathbf{h}_\mathrm{k}}{\mathrm{d}t}$ and $\frac{\mathrm{d}\lambda_\mathrm{r}}{\mathrm{d}t}$ of the shape model.

Transfer to the geometric state space. The approach above is formulated in terms of an unconstrained $\mathcal{H}$-representation for the kernel polytope. It can be transferred to the $\mathbf{h}_\mathrm{C}$-vectors of a suitable $\mathcal{H}_\mathrm{C}$-representation when the constraint $\mathbf{M}_\mathrm{C}\mathbf{1} = \mathbf{0}$ is satisfied. In applications it will typically hold that each component of the $\mathbf{h}_\mathrm{C}$-vector directly represents multiple facet distances so that equation 5.20 can be reformulated to:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mu^{\text{mean width}}\left(\mathbf{h}_{\mathrm{C,k}}, \lambda_\mathrm{r}\right) = \left(2 - \left(\left.\frac{\partial \mu^{\text{mean width}}\left(\mathbf{h}_{\mathrm{C,k}}\right)}{\partial \mathbf{h}'_{\mathrm{C,k}}}\right|_{\mathbf{h}_{\mathrm{C,k}}} \cdot \mathbf{1}\right)\right)\frac{\mathrm{d}\lambda_\mathrm{r}}{\mathrm{d}t}. \tag{5.22}$$

Necessity of a second mechanism. The decrease of the particle mean width must also be realized for fully rounded particles. In this case, the kernel polytope cannot further reduce in size so that a second effect must be introduced and coupled to the effect of pure rounding. This second effect will be realized by a shape preserving decrease in size and the required considerations for the coupling of the two mechanisms are analogue to the work of Briesen [2009a].

Shape preservation. The preservation of the particle shape $C(\mathbf{h}_{\mathrm{C,k}}, \lambda_{\mathrm{r}})$ raises the constraint:

$$0 = \frac{\mathrm{d}\left(\frac{\mathbf{h}_{\mathrm{k}}}{\lambda_{\mathrm{r}}}\right)}{\mathrm{d}t} = \frac{1}{\lambda_{\mathrm{r}}}\frac{\mathrm{d}\mathbf{h}_{\mathrm{k}}}{\mathrm{d}t} - \frac{\mathbf{h}_{\mathrm{k}}}{\lambda_{\mathrm{r}}^2}\frac{\mathrm{d}\lambda_{\mathrm{r}}}{\mathrm{d}t} \tag{5.23}$$

$$\frac{\mathrm{d}\mathbf{h}_{\mathrm{k}}}{\mathrm{d}t} = \frac{\mathbf{h}_{\mathrm{k}}}{\lambda_{\mathrm{r}}}\frac{\mathrm{d}\lambda_{\mathrm{r}}}{\mathrm{d}t}. \tag{5.24}$$

This constraint can be applied to the time derivative of the mean width and results in:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mu^{\text{mean width}}\left(\mathbf{h}_{\mathrm{k}}, \lambda_{\mathrm{r}}\right) = \left(\mu^{\text{mean width}}\left(\frac{\mathbf{h}_{\mathrm{k}}}{\lambda_{\mathrm{r}}}\right) + 2\right)\frac{\mathrm{d}\lambda_{\mathrm{r}}}{\mathrm{d}t} \tag{5.25}$$

or:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mu^{\text{mean width}}\left(\mathbf{h}_{\mathrm{C,k}}, \lambda_{\mathrm{r}}\right) = \left(\mu^{\text{mean width}}\left(\frac{\mathbf{h}_{\mathrm{C,k}}}{\lambda_{\mathrm{r}}}\right) + 2\right)\frac{\mathrm{d}\lambda_{\mathrm{r}}}{\mathrm{d}t}. \tag{5.26}$$

Similar to the effect of pure rounding, equations 5.24 and 5.26 are used to transfer a rate for the mean width into rates of the shape model.

Coupling of pure rounding and shape preserving rates. Modeling the dynamics of attrition, a size changing rate $g_{\text{attrition}}$ in terms of the mean width of the crystal is assumed. This rate is realized on the one hand by the effect of pure rounding that results in a change of the mean with, denoted by $g_{\text{attrition}}^{\text{rounding}}$ and on the other hand by a change in the mean width $g_{\text{attrition}}^{\text{const. shape}}$ for which the shape is preserved. To guarantee the given overall size changing rate, the constraint:

$$g_{\text{attrition}} = g_{\text{atttrition}}^{\text{const. shape}} + g_{\text{attrition}}^{\text{rounding}} \tag{5.27}$$

applies and the ratio:

$$q_{\text{attrition}}\left(\mu^{B3}\right) = \frac{g_{\text{attrition}}^{\text{rounding}}}{g_{\text{attrition}}^{\text{const. shape}}}. \tag{5.28}$$

is used to realize the pure rounding and shape preserving portions of the overall shape change. For $q_{\text{attrition}} \to \infty$, the particle size changes while it is rounded or de-rounded, only and for $q_{\text{attrition}} = 0$, the particle shape is preserved. This ratio is dependent on the volume based additive roundness since a fully rounded particle cannot become any more rounded so that the overall changing rate must be realized by the shape preserving effect. This observation implies that $q_{\text{attrition}}(1) = 0$ must hold. Additionally demanded is $q_{\text{attrition}}(0) < \infty$ which implies that even the facets of a fully angular shape are slightly abraded. Furthermore, the function $q_{\text{attrition}}\left(\mu^{B3}\right)$ should be monotonically increasing so that $\frac{\mathrm{d}q_{\text{attrition}}(\bar{\mu}_B)}{\mathrm{d}\bar{\mu}_B} \leq 0$ holds. These limitations to the ratio $q_{\text{attrition}}\left(\mu^{B3}\right)$ were already discussed by Briesen [2009a] who also proposed a suitable parametreized function that is adopted for this work:

$$q_{\text{attrition}}\left(\mu^{B3}\right) = k_{\max}\frac{e^{a_{\text{slope}}\mu^{B3}} - e^{a_{\text{slope}}}}{1 - e^{a_{\text{slope}}}}. \tag{5.29}$$

Some examples for this ratio are plotted in figure 5.7. The parameter $k_{\max}$ determines the maximum ratio that is realized for $\mu^{B3} = 0$. The bold lines in figure 5.7 apply

$k_{\max} = 1$ while the thin lines apply $k_{\max} = 0.2$. The parameter a_{slope} determines the slope of the ratio function. For $a_{\text{slope}} < 0$, the ratio reduces quickly for small roundness values μ^{B3} as it is shown by the light blue lines in figure 5.7. For $a_{\text{slope}} > 0$, the ratio remains high with low roundness values μ^{B3} as it is shown by the blue lines. The black lines use $a_{\text{slope}} = 10^{-4}$ and demonstrate that the limiting case $a_{\text{slope}} = 0$ for which equation 5.29 is not defined results in:

$$q_{\text{attrition}}\left(\mu^{B3}\right) = k_{\max}\left(1 - \mu^{B3}\right). \tag{5.30}$$

Summary. Assembling equations 5.3 and 5.20 for the rounding effect, equations 5.24 and 5.26 for the shape preserving rates and equations 5.27 and 5.28 for the coupling of the effects results in a dynamic model for attrition:

$$\frac{\mathrm{d}\mathbf{h}_{\mathrm{R}}}{\mathrm{d}t} = \begin{bmatrix} \frac{\mathbf{h}_{\mathrm{C,k}}}{\lambda_{\mathrm{r}}} & -\mathbf{1} \\ 1 & 1 \end{bmatrix} \begin{pmatrix} \frac{1}{2+\mu^{\text{mean width}}\left(\frac{\mathbf{h}_{\mathrm{C,k}}}{\lambda_{\mathrm{r}}}\right)} \\ \frac{q_{\text{attrition}}\left(\mu^{B3}\right)}{2-\left.\frac{\partial \mu^{\text{mean width}}\left(\mathbf{h}_{\mathrm{C,k}}\right)}{\partial \mathbf{h}'_{\mathrm{C,k}}}\right|_{\mathbf{h}_{\mathrm{C,k}}} \cdot \mathbf{1}} \end{pmatrix} \frac{g_{\text{attrition}}}{q_{\text{attrition}}\left(\mu^{B3}\right)+1}. \tag{5.31}$$

The overall decrease in mean width ($g_{\text{attrition}}$) might be computed by a mechanistic model or identified experimentally. Additionally, the parameters for the ratio function $q_{\text{attrition}}\left(\mu^{B3}\right)$ must be set. Note also that any initial condition approaches $\mathbf{h}_{\mathrm{R}} = \mathbf{0}$ for $g_{\text{attrition}} < 0$. Initially, the kernel polytope might decrease while the radius λ_{r} increases. Then, while the particle approaches a well rounded shape with $\mu^{B3} \approx 1$, its shape is preserved so that, neccessarily, $\mathbf{h}_{\mathrm{R}} = \mathbf{0}$ is approached. It follows that only the validity of the vector $\mathbf{h}_{\mathrm{C,k}}$ must be considered during the simulation and a particle disappears when its $\mathbf{h}_{\mathrm{R}}$-vector is close to zero.

Case study. The derived equations are implemented and evaluated to demonstrate the effect of the ratio function $q_{\text{attrition}}\left(\mu^{B3}\right)$ on the evolution of roundness. Therefore, a single particle with a mean width of 1 decreases constantly in mean width ($g_{\text{attrition}} = -1$) until it disappears. Figures 5.9 and 5.10 show in the left the initial particle shape and snapshots from equidistant time steps between 0 and 0.5. Figure 5.8 shows the volume based and mean width based additive roundness over time where the style of the lines matches the styles from figure 5.7 that contains the applied ration functions $q_{\text{attrition}}$. In particular, figures 5.9 and 5.10 illustrate that the parameters of the ratio function can be well adjusted to describe different rates at which the particles become rounded.

5.2.3 Growth Dynamics

Motivation. Angular particles typically originate from the anisotropic growth of crystals or from breakage for general particles. Therefore, this subsection discusses a dynamic modeling approach that covers the de-rounding of a rounded particle due to growth. Breakage is not considered because it represents a discontinuous event with respect to the geometrical state space. Given the attrition dynamics in the previous subsection and the growth dynamics in this subsection, it will be possible in future to apply new model based approaches to study attrition kinetics or growth kinetics of high indexed facets.

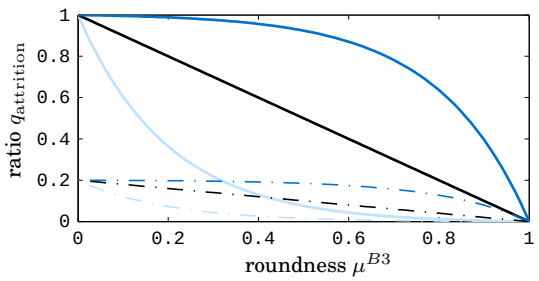

Figure 5.7.: Attrition rounding ratio $q_{\text{attrition}}(\bar{\mu}_{\text{B}})$ plotted with different parameters: bold lines use $k_{\max} = 1$, dash-dotted lines use $k_{\max} = 0.2$, blue lines use $a_{\text{slope}} = 5$, light blue lines use $a_{\text{slope}} = -5$ and black lines use $a_{\text{slope}} = 10^{-4}$

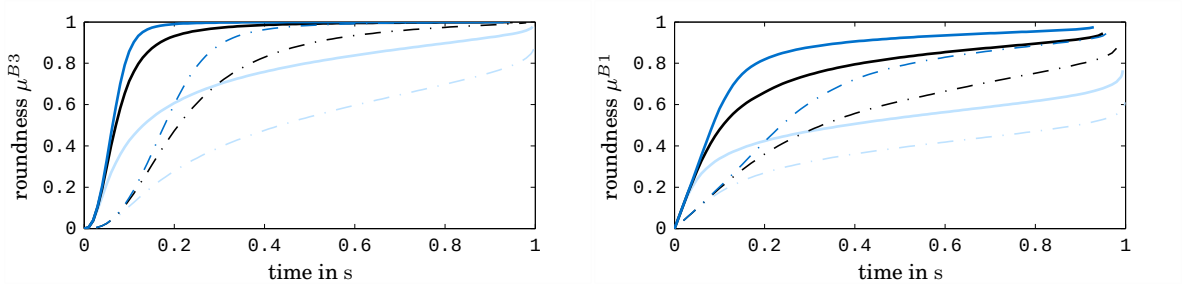

Figure 5.8.: Time evolution of the volume based additive roundness (left) and mean width based additive roundness (right) during a hypothetical attrition according to the ratio functions in figure 5.7 (line styles match the applied ratio function)

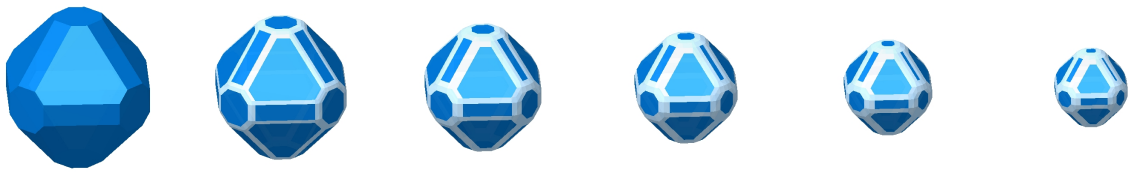

Figure 5.9.: Shape snapshots in equidistant times from 0 to 0.5 for the slowest rounding with $a_{\text{slope}} = -5$ and $k_{\max} = 0.2$

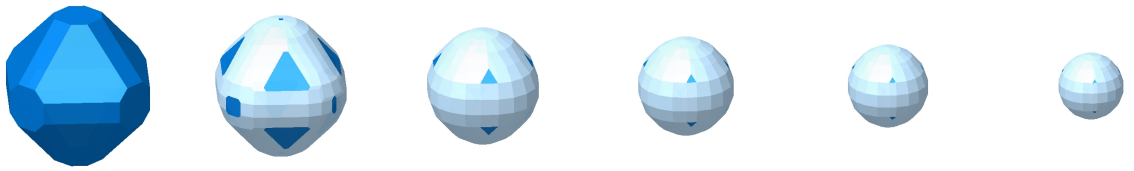

Figure 5.10.: Shape snapshots in equidistant times from 0 to 0.5 for the fastest rounding with $a_{\text{slope}} = 5$ and $k_{\max} = 1$

Modeling and assumptions. Growth rates for the morphologically most important facets are typically known when a corresponding shape model is utilized. The growth of all other facets that are exposed due to the rounded edges and vertices are typically unknown. Analogue to the attrition model, the growth of a rounded particle is split into two mechanisms. This approach guarantees that the known rates of the crystal facets are properly realized. The first mechanism implements the known growth rates of the facets by applying them to the kernel polytope:

$$\frac{\mathrm{d}\mathbf{h}_{\mathrm{C,k}}}{\mathrm{d}t} = \mathbf{g}_{\mathrm{growth}} \tag{5.32}$$

while the radius λ_{r} remains constant:

$$\frac{\mathrm{d}\lambda_{\mathrm{r}}}{\mathrm{d}t} = 0. \tag{5.33}$$

This ensures that the facets of the circumscribing polytope grow properly by: $\frac{\mathrm{d}\mathbf{h}_{\mathrm{c}}}{\mathrm{d}t} = \mathbf{g}_{\mathrm{growth}}$. This mechanism also implies a displacement of all other facets according to the displacement of the edges and vertices of the kernel polytope. The second mechanism covers the pure de-rounding of the shape and is realized according to equations 5.3 and 5.20. Since this mechanism assumes $\frac{\mathrm{d}\mathbf{h}_{\mathrm{c}}}{\mathrm{d}t} = \mathbf{0}$, the growth of the modeled facets remains unchanged. However, unknown is the increase in mean width that shall be realized by the de-rounding mechanism. It is denoted by $g_{\mathrm{growth}}^{\mathrm{rounding}}$. Since only one degree of freedom is available for all possible facets of the rounded shape portions, a reasonable approach is to average the known growth rates to obtain the required rate. Here, it is assumed that the additional de-rounding effect realizes an additional increase of the mean width, proportional to the increase in mean width by the pure growth mechanism (equations 5.32 and 5.33). This results in:

$$g_{\mathrm{growth}}^{\mathrm{rounding}} = q_{\mathrm{growth}} \left.\frac{\partial \mu^{\mathrm{mean\,width}}\left(\mathbf{h}'_{\mathrm{k}}\right)}{\partial \mathbf{h}'_{\mathrm{k}}}\right|_{\mathbf{h}_{\mathrm{k}}} \cdot \mathbf{g}_{\mathrm{growth}} \tag{5.34}$$

where q_{growth} is the proportionality constant. While a constant $q_{\mathrm{growth}} = 0$ implies that no additional de-rounding mechanism is realized, the overall roundness still decreases. In this case, the radius λ_{r} remains constant while the overall particle size increases so that the radius becomes negligible. In contrast, a positive constant $q_{\mathrm{growth}} > 0$ implies a decreasing radius λ_{r} that approaches $\lambda_{\mathrm{r}} = 0$. This boundary must not be crossed so that a corresponding validity mapping is required. However, it is not reasonable to simply correct $\frac{\mathrm{d}\lambda_{\mathrm{r}}}{\mathrm{d}t}$ to $\frac{\mathrm{d}\lambda_{\mathrm{r}}}{\mathrm{d}t} = 0$, independently of the $\mathbf{h}_{\mathrm{C,k}}$ rates. An appropriate mapping direction according to subsection 3.2.3 retains the facet distances of the circumscribing polytope. This implies to retain $\mathbf{h}_{\mathrm{c}}$ so that according to equation 5.3 a mapping direction:

$$\mathbf{h}_{\mathrm{R},\mapsto} = \begin{pmatrix} -\mathbf{1} \\ 1 \end{pmatrix} \tag{5.35}$$

must be applied for the validity mapping[2].

[2]The direction $\mathbf{h}_{\mathrm{R},\mapsto}$ can be substituted for the direction $\mathbf{h}_{\mathrm{C},\mapsto}$ that is used in subsection 3.2.3. Likewise, the facet normal matrix of the facet validity cone for λ is referenced by $\mathbf{A}_{\mathrm{FV}(B)}$ and corresponds to $\mathbf{A}_{\mathrm{FV}(i)}$.

Case study. To demonstrate the dynamic model and to illustrate the influence of the constant q_{growth}, the following simulation case study was conducted. It simulates potash alum in water using the facets $\{111\}$, $\{110\}$ and $\{100\}$ for which the growth rates and crystal data is taken from Ma and Wang [2008] and Nollet et al. [2006]. They are summarized in table 5.1. The initial crystal population is given directly in its discretized form by 100 logarithmically distributed sample points with $i \in [1, \ldots, 100]$:

$$\mathbf{h}_{\mathrm{R},i} = \begin{pmatrix} 0.01 \\ 0.01 \\ 0.01 \\ 1 \end{pmatrix} 10^{-1+\frac{2}{99}(i-1)}\,\mu\mathrm{m}. \tag{5.36}$$

Each sample point represents 2×10^7 particles so that a total mass of $1.2\,\mathrm{g}$ is obtained. The vessel volume is set to $V_{\text{vessel}} = 1 \times 10^{-3}\,\mathrm{m}^3$ water which results in a particle volume fraction of about 0.07%. The initial solution has a temperature of $60\,^\circ\mathrm{C}$ and is saturated. It is cooled down to $30\,^\circ\mathrm{C}$ within one hour. The supersaturation is given by:

$$s(t) \quad = \quad \frac{(V_{\text{cry},0} - V_{\text{cry}}(t))\,\frac{\varrho_{\text{potash}}}{m_{\text{mol,potash}}} + V_{\text{vessel}}\,c_{\text{sol,potash}}(60\,^\circ\mathrm{C})}{V_{\text{vessel}}\,c_{\text{sol,potash}}(T(t))} - 1 \tag{5.37}$$

where $T(t)$ is the current temperature, $V_{\text{cry},0}$ is the initial crystal volume and $V_{\text{cry}}(t)$ is the current crystal volume. Three simulation runs were executed using $q_{\text{growth}} = 0$, $q_{\text{growth}} = 0.01$ and $q_{\text{growth}} = 0.1$.

Figure 5.11 shows the supersaturation (left) and the crystal volume (right) over time. The initial total crystal surface is not sufficiently large to absorb the supersaturation that is created by the temperature gradient so that a peak can be observed in the supersaturation plot. This behavior is also evident by the total crystal volume in the right graph. With regard to the different values of the constant q_{growth}, no significant differences can be observed. Figure 5.12 shows the roundness and mean width evolution for three selected crystal trajectories. The left figures represent the smallest initial particle, the middle column a medium sized particle and the right column the largest particle of the population. Again, the constant q_{growth} has no significant effect on the resulting crystal sizes. Only large initial crystals tend to grow slightly larger for larger values of q_{growth}. In contrast, the plots of the roundness μ^{B3} show that different degrees of roundness are obtained. In general, small particles become rounded much faster than large particles. The radius λ_{r} that has to vanish for particles in order to become fully angular scales with its size while the changing rate is the same for all particles, which explains this observation. Secondly, even a small constant $q_{\text{growth}} = 0.1$ leads to a fast disappearance of rounded particles.

Table 5.1.: Crystal data for potash alum

name	definition
crystal system	cubic, $m\bar{3}$, $a = 12.517\,\text{Å}$
crystal density	$\varrho_{\text{potash}} = 1753\,\frac{\text{kg}}{\text{m}^3}$
molar mass	$m_{\text{mol,potash}} = 258.2063\,\frac{\text{kg}}{\text{mol}}$
solubility in water	$c_{\text{sol,potash}}(T) = e^{\frac{12.1\,\text{K}}{T}+10.47\ln\left(\frac{T}{\text{K}}\right)-65.73} \times 10^5\,\frac{\text{mol}}{\text{m}^3}$
growth rate for $\{111\}$ facets	$g_{\{111\}}(s) = 7.753\,s^{1.5} \times 10^{-7}\,\frac{\text{m}}{\text{s}}$
growth rate for $\{110\}$ facets	$g_{\{110\}}(s) = 1.124\,s^{1.5} \times 10^{-6}\,\frac{\text{m}}{\text{s}}$
growth rate for $\{100\}$ facets	$g_{\{111\}}(s) = 1.744\,s^{1.5} \times 10^{-6}\,\frac{\text{m}}{\text{s}}$

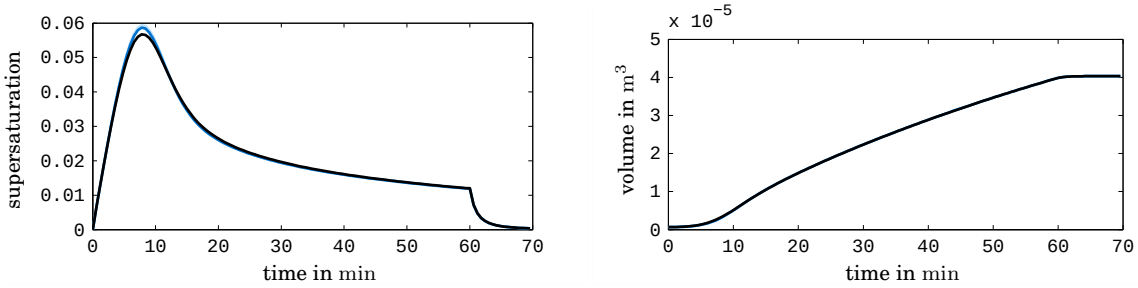

Figure 5.11.: Supersaturation (left) and total crystal mass (right) over time.

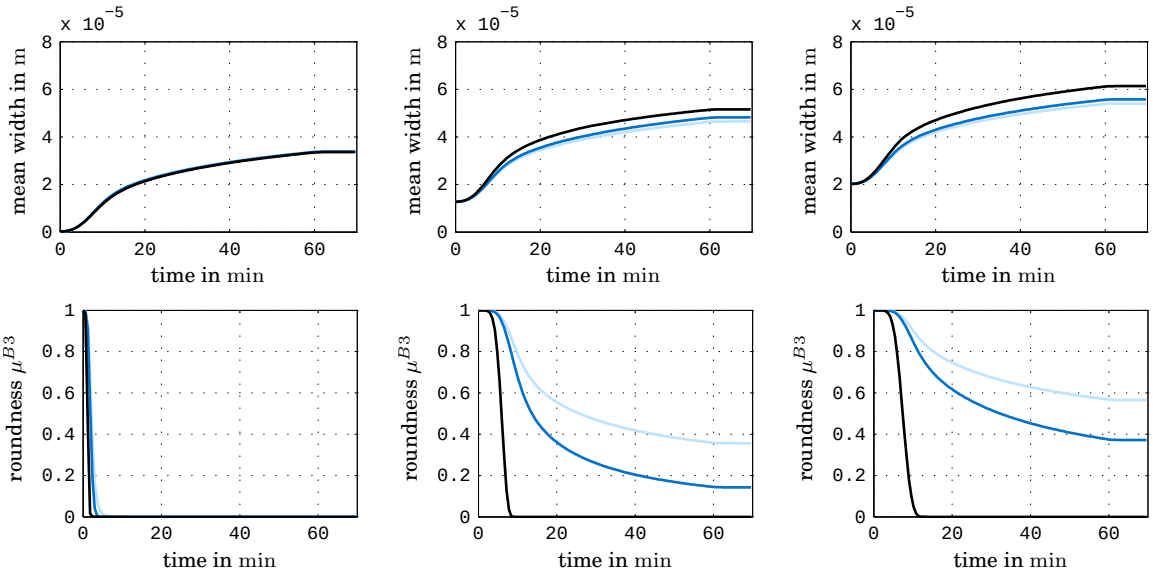

Figure 5.12.: Mean width (top row) and roundness (bottom row) over time for the smallest particle (left), medium sized particle (middle) and large particle (right) and the three simulation runs using $q_{\text{growth}} = 0$ (light blue lines), $q_{\text{growth}} = 0.01$ (blue lines) and $q_{\text{growth}} = 0.1$ (black lines)

5.3 Measuring Attrition Processes

5.3.1 Identification Algorithm

Motivation. Tracking the roundness of particles in a crystallization process would be valuable to study attrition kinetics as it is discussed in the introduction to this chapter. Therefore, this section presents an algorithm that identifies the size ($\mu^{\text{mean width}}$) and roundness (μ^{B1}) from particle projection images. The algorithm is designed as a shape identification procedure and returns also the $\mathcal{H}$-representation of the projected kernel polytope. This information could be utilized in an existent shape identification technique [Schorsch et al., 2012] to obtain the thickness and length of rod-like kernel polytopes or the height, width and depth of cuboid-like kernel polytopes. This would, in principle, allow to additionally track the particle volume and surface area.

Thanks to the group of Prof. Mazzotti (ETH Zürich, Stefan Schorsch in particular), a stereoscopic particle imaging setup was available to test the algorithm with real measurement data. However, only separate projections are discussed throughout this section and the required detail to match both perpendicular projections from the measurement setup is given in appendix A.1.4. Details of the measurement setup and the experiment setup are given in subsection 5.3.3 while the assumptions to the input data and the algorithm itself are presented here.

Algorithm output. Given that the volume based additive roundness is the only introduced roundness definition that guarantees a roundness value of 1 for fully rounded particles, it would be the most appropriate measure. Additionally, the particle volume is a commonly used size measure. However, volume information is difficult to obtain from 2-dimensional imaging data so that the primary result of the presented algorithm is the mean width of the particle as a size measure and the mean width based additive roundness.

Input data. The assumed input data comprises the coordinates of points on the boundary of the projected particle like it is shown by the white dots in the left of figure 5.13. They are assumed to be reasonably well uniformly distributed over the boundary of the projected particle. The total number of points is denoted by n_{point} while $\mathbf{x}_i$ denotes their representation in an Cartesian coordinate system which has its origin in the interior of the particle boundary.

Projected shape model. The particle is assumed to follow the shape model from subsection 5.1.2. Since the projection of a Minkowski sum equals the sum of their projected summands [Schneider, 2008], equation 5.2 results in:

$$\Pi C(\mathbf{h}_{\text{k}}, \lambda_{\text{r}}) = \Pi C(\mathbf{h}_{\text{k}}) + \Pi\left(\lambda_{\text{r}} B\right). \tag{5.38}$$

where the symbol Π denotes the orthogonal projection of the subsequently listed body as an operator. Equation 5.38 can be rewritten to:

$$\Pi C(\mathbf{h}_{\text{k}}, \lambda_{\text{r}}) = P(\mathbf{h}_{\text{k},\Pi}) + \lambda_{\text{r},\Pi} D \tag{5.39}$$

where $P(\mathbf{h}_{\text{k},\Pi})$ is a 2-dimensional kernel polytope that is given in $\mathcal{H}$-representation with the facet normal matrix $\mathbf{A}_{\text{k},\Pi}$ and the facet distance vector $\mathbf{h}_{\text{k},\Pi}$ while $\lambda_{\text{r},\Pi} D$ is a disk with radius $\lambda_{\text{r},\Pi}$. Note the specific projection Π being added to the index for these new shape parameters to distinguish the symbols from the 3-dimensional shape model.

Algorithm overview. Figure 5.13 shows in the left a projection of a rounded particle together with the data points plotted in white. From visual impression, it is difficult to identify the parameters $\mathbf{h}_{\mathrm{k},\Pi}$ and $\lambda_{\mathrm{r},\Pi}$ of the projected shape model directly. In contrast to that, it is relatively simple to identify a well fitting circumscribing polytope by identifying lines in the data points. This step is performed by a Hough transformation whose details are described later while the result is illustrated in the middle of figure 5.13. The corresponding polytope is described in $\mathcal{H}$-representation by a facet normal matrix $\mathbf{A}_{\mathrm{c},\Pi}$ and the facet distance vector $\mathbf{h}_{\mathrm{c},\Pi}$. Given the circumscribing polytope and assuming an yet unknown radius $\lambda_{\mathrm{r},\Pi}$, the kernel polytope is known based on the same $\mathcal{H}$-representation ($\mathbf{A}_{\mathrm{k},\Pi} = \mathbf{A}_{\mathrm{c},\Pi}$) while using the facet distance vector:

$$\mathbf{h}_{\mathrm{k},\Pi} = \mathbf{h}_{\mathrm{c},\Pi} - \lambda_{\mathrm{r},\Pi} \tag{5.40}$$

analogue to equation 5.3. To identify the radius $\lambda_{\mathrm{r},\Pi}$, two principal approaches are proposed. The first alternative resembles the roundness definition and measurement procedure that is given by Wadell [1932]. It identifies the roundness for each corner separately so that it can be expected that the algorithm also performs reasonably well for particle aggregates. The second alternative represents a typical least squares approach and aims to identify the radius $\lambda_{\mathrm{r},\Pi}$ so that the resulting shape fits best to the given data points. A fitted sample shape is shown in the right of figure 5.13. The kernel polytope is outlined by the light lines in the center while the circles indicate the vertices of the kernel polytope and illustrate how the rounded corners of the overall shape are created.

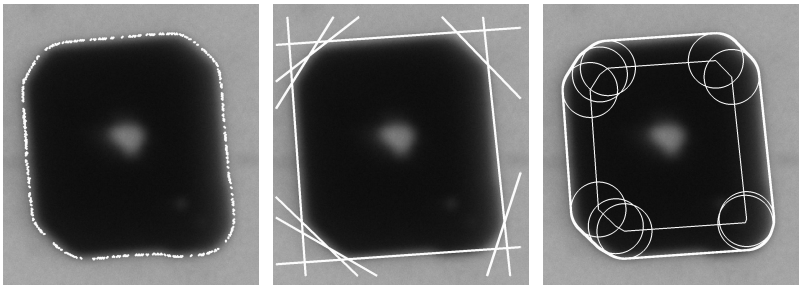

Figure 5.13.: Overview of the identification algorithm based on a sample shape with the data points $\mathbf{x}_i$ in white (left), the identified lines that are featured by the input points (middle) and the fitted shape (bold white line in the right)

Line detection. The lines that are featured by the sample points $\mathbf{x}_i$ are identified following the principles of a Hough transformation [Hough, 1962, Illingworth and Kittler, 1988, Burger and Burge, 2009]. The key element of a Hough transformation is a parameterized form of the image feature that needs to be detected. For the present case of lines to be detected, the parameterization is given by an angle $\bar{\varphi}$ and a distance from the origin $\bar{\rho}$. The line is then determined by all points $\mathbf{x}$ that fulfill:

$$\bar{\rho} = \left\langle \mathbf{x}, \begin{pmatrix} \cos\bar{\varphi} \\ \sin\bar{\varphi} \end{pmatrix} \right\rangle. \tag{5.41}$$

In a next step, the parameter space is discretized and corresponding bin values are introduced. The angles are discretized by the sequence:

$$\bar{\varphi}_0 = 0 \qquad \bar{\varphi}_{j+1} = \bar{\varphi}_j + \Delta_{\bar{\varphi}} \qquad \bar{\varphi}_j \leq 2\pi \tag{5.42}$$

where the step size $\Delta_{\bar{\varphi}} = 2°$ determines the accuracy of the orientation for each line. The distance $\bar{\rho}$ is discretized based on the minimum and maximum polar distances of the data points:

$$\rho_{\min} = \min_i \sqrt{\langle \mathbf{x}_i, \mathbf{x}_i \rangle} \tag{5.43}$$

$$\rho_{\max} = \max_i \sqrt{\langle \mathbf{x}_i, \mathbf{x}_i \rangle} \tag{5.44}$$

and according to the sequence:

$$\bar{\rho}_0 = 0.95\,\rho_{\min} \qquad \bar{\rho}_{k+1} = \bar{\rho}_k + \Delta_{\bar{\rho}} \qquad \bar{\rho}_k \leq 1.05\,\rho_{\max}. \tag{5.45}$$

The factors 0.95 and 1.05 are suitable for most applications while $\Delta_{\bar{\rho}} = 1$ (pixel) determines the accuracy of the distance between the line and the origin.[3] The bin values are denoted by $b_{(j,k)}$ where j denotes the discretized angle $\bar{\varphi}_j$ and k denotes the discretized distance $\bar{\rho}_k$. Each sample point $\mathbf{x}_i$ supports different lines so that each bin value $b_{(j,k)}$ is incremented by one for each sample point $\mathbf{x}_i$ that supports the corresponding parameter combination $\bar{\varphi}_j$ and $\bar{\rho}_k$ which is computed by:

$$\begin{aligned} \left\langle (\cos\bar{\varphi}_j,\, \sin\bar{\varphi}_j)^T,\, \mathbf{x}_i \right\rangle &\geq \bar{\rho}_k - \frac{1}{2}\Delta\bar{\rho} \\ \left\langle (\cos\bar{\varphi}_j,\, \sin\bar{\varphi}_j)^T,\, \mathbf{x}_i \right\rangle &< \bar{\rho}_k + \frac{1}{2}\Delta\bar{\rho}. \end{aligned} \tag{5.46}$$

The bin values $b_{(j,k)}$ can be interpreted as a gray scale image like the one shown in figure 5.14. The dark spots indicate regions of high bin values.[4] These spots also indicate the parameter combinations of the longest lines that are featured by the data points $\mathbf{x}_i$. For the algorithm in this subsection, the evaluation of these bin values deviates slightly from typical applications of a Hough transformation. First, since the boundary of a reasonably convex shape is identified, each angle $\bar{\varphi}_j$ can only feature a single line. Secondly, in common applications only such lines are detected that are featured by some minimum number of points $\mathbf{x}_i$ and/or that correspond to a limited number of local maxima in the bin values. This only identifies particularly long lines while, for this application, the length of the lines is not defining the importance of the corresponding facet. Instead, parameter combinations $\bar{\varphi}_j$ and $\bar{\rho}_k$ are chosen when no bin value $b_{(j',k')} \geq b_{(j,k)}$ exists where j' is constrained so that $\bar{\varphi}_j - \frac{1}{2}\Delta_{\varphi,\text{sep}} \leq \bar{\varphi}_{j'} \leq \bar{\varphi}_j + \frac{1}{2}\Delta_{\varphi,\text{sep}}$ holds [Neubeck and van Gool, 2006, Burger and Burge, 2009]. The parameter $\Delta_{\varphi,\text{sep}} = 15°$ separates local minima and implies that the angles $\bar{\varphi}_j$ for the identified lines differ at least about the value of $\Delta_{\varphi,\text{sep}}$. In summary, the detected lines yield the $\mathcal{H}$-representation of the circumscribed polytope with the facet normals:

$$\mathbf{a}_{\text{c},\Pi,l} = (\cos\bar{\varphi}_j,\, \sin\bar{\varphi}_j)^T \tag{5.47}$$

[3] The value 1.05 only considers issues that can be created by a coarse discretization. No line can be farther away than $\rho_{\max}$ and still be featured by any point $\mathbf{x}_i$. The same applies for the factor 0.95.

[4] Note that the upper outline in figure 5.14 represents the support function for the data points $\mathbf{x}_i$.

and the facet distances:

$$h_{\mathrm{c},\Pi,l} = \bar{\rho}_k \tag{5.48}$$

as it is illustrated in the middle of figure 5.13.

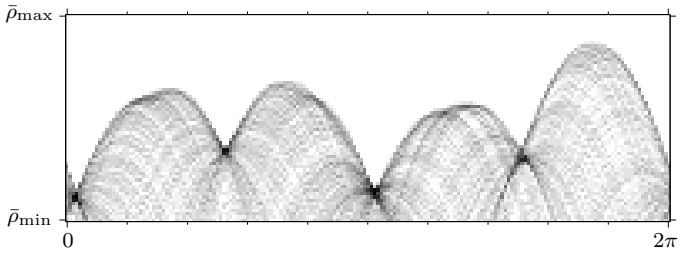

Figure 5.14.: Visualization of the bin values $b_{(j,k)}$ from the Hough transformation of the points in figure 5.13 with the maximum value in black and values $b_{(j,k)} = 0$ in white

Signed distances. The following two approaches to identify the radius $\lambda_{\mathrm{r},\Pi}$ require a definition for the goodness of fit between the shape model (equation 5.39) and the data points $\mathbf{x}_i$. Such a measure can be derived from the signed distances $\delta(\mathbf{x}_i)$ of each data point from the shape boundary. If the point $\mathbf{x}_i$ is inside of the kernel polytope, the distance is given by:

$$\delta(\mathbf{x}_i) = \min_l \left(\mathbf{a}_{\mathrm{k},\Pi,l}\mathbf{x}_i - h_{\mathrm{k},\Pi,l}\right) - \lambda_{\mathrm{r},\Pi} \tag{5.49}$$

which can be computed directly. If the point $\mathbf{x}_i$ resides outside of the kernel polytope, which is typically the case, the distance is defined by the constrained quadratic optimization problem:

$$\begin{aligned} \delta(\mathbf{x}_i) &= \min_{\mathbf{x}} \langle \mathbf{x},\, \mathbf{x}_i \rangle - \lambda_{\mathrm{r},\Pi} \\ \mathbf{A}_{\mathrm{k},\Pi}\mathbf{x} &\leq \mathbf{h}_{\mathrm{k},\Pi}. \end{aligned} \tag{5.50}$$

While this problem can be solved by almost any optimization algorithm, the underlying geometry was further exploited. First, one constraint is necessarily active. Otherwise, the point $\mathbf{x}_i$ resides inside the kernel polytope and equation 5.49 applies. This active constraint $\mathbf{a}_{\mathrm{k},\Pi,l}^T\mathbf{x} \leq h_{\mathrm{k},\Pi,l}$ is known by the facet l that maximizes $\mathbf{a}_{\mathrm{k},\Pi,l}^T\mathbf{x}_i - h_{\mathrm{k},\Pi,l}$. Thus, the resulting distance is:

$$\delta(\mathbf{x}_i) = \mathbf{a}_{\mathrm{k},\Pi,l}\mathbf{x}_i - h_{\mathrm{k},\Pi,l} - \lambda_{\mathrm{r},\Pi} \tag{5.51}$$

when the sample point $\mathbf{x}_i$ can be projected perpendicular on the facet l so that the projected point $\mathbf{x}_i'$ resides on the boundary of the kernel polytope: $\mathbf{A}_{\mathrm{k},\Pi}\mathbf{x}_i' \leq \mathbf{h}_{\mathrm{k},\Pi}$. Otherwise, the resulting distance is:

$$\delta(\mathbf{x}_i) = \langle \mathbf{x}_{\mathrm{E},m},\, \mathbf{x}_i \rangle - \lambda_{\mathrm{r},\Pi} \tag{5.52}$$

where $\mathbf{x}_{\mathrm{E},m}$ is one of the two vertices[5] that are adjacent to the facet l, whichever is closest. The distance $\delta(\mathbf{x}_i)$ is negative when the point $\mathbf{x}_i$ is an element of the shape $C(\mathbf{h}_{\mathrm{k},\Pi}, \lambda_{\mathrm{r},\Pi})$ and positive, otherwise.

[5] extreme points

Error of fitted shape. A typical approach for an error measure is the root of the mean squared distances $\delta(\mathbf{x}_i)$:

$$\delta_{\text{mean squared}} = \sqrt{\frac{1}{n_{\text{point}}} \sum_i \delta(\mathbf{x}_i)^2}. \tag{5.53}$$

This error measure is well suitable as an objective function to determine a fitting shape $C(\mathbf{h}_{\text{k},\Pi}, \lambda_{\text{r},\Pi})$ to the points $\mathbf{x}_i$. However, its informative value is insufficient to determine problematic input data. Bumps, like the one in the bottom left of figure 5.15 indicate nearby background noise, nearby bubbles or aggregates that might yield unreliable results in the line detection step. Such features of the input data are suppressed which makes it a good objective function in the first place. To identify such features, a second error measure is introduced by:

$$\delta_{\text{profile height}} = \max_i \delta(\mathbf{x}_i) - \min_i \delta(\mathbf{x}_i). \tag{5.54}$$

The profile height $\delta_{\text{profile height}}$ is visualized in figure 5.15 and the minimum and maximum components in equation 5.54 correspond to the two bold white lines. The inner shape corresponds to $C(\mathbf{h}_{\text{k},\Pi} + \min_i \delta(\mathbf{x}_i),\, \lambda_{\text{r},\Pi})$ and no point $\mathbf{x}_i$ resides in its interior. The outer shape corresponds to $C(\mathbf{h}_{\text{k},\Pi} + \max_i \delta(\mathbf{x}_i),\, \lambda_{\text{r},\Pi})$ and all points $\mathbf{x}_i$ are contained in that shape. This error measure is used as the general goodness of fit because it conveys information on small bumps in the particle outline. Note that, despite their use as an error measure, the root of the mean squares $\delta_{\text{mean squared}}$ and the profile height $\delta_{\text{profile height}}$ are also descriptors for the roughness of the surface structure given that the shape model is consistent with the particle shape and given that measurement noise is negligible. The square height:

$$\delta_{\text{square height}} = \max_i \delta(\mathbf{x}_i)^2 + \min_i \delta(\mathbf{x}_i)^2 \tag{5.55}$$

is introduced as a last error measure to yield an error measure that can be used as an objective function to fit a shape to the input points $\mathbf{x}_i$. The profile height is not suited for this because any new radius $\lambda'_{\text{r},\pi}$ with:

$$\lambda_{\text{r},\pi} + \min_i \delta(\mathbf{x}_i) \leq \lambda'_{\text{r},\pi} \leq \lambda_{\text{r},\pi} + \max_i \delta(\mathbf{x}_i) \tag{5.56}$$

results in the same profile height. On the contrary, the square height becomes minimal when the shape is well centered between the inner and outer limiting shape.

Determining the radius $\lambda_{\text{r},\Pi}$. The first approach to identify the radius $\lambda_{\text{r},\Pi}$ resembles the definition and measurement procedure by Wadell [1932] who identifies corners in the particle projection and fits the arc of a circle to each corner. His approach ensures that about 3 to 5 corners are used to determine the mean curvature radius of the corners. For the present application, the vertices of the circumscribing polytope $P(\mathbf{h}_{\text{c},\Pi})$ are used to indicate the corners of the particle projection. A suitable averaging of the radii $\lambda_{\text{r},\Pi,j}$ for each vertex j is then given by:

$$\lambda_{\text{r},\Pi} = \sum_j \frac{\alpha_j}{2\pi} \lambda_{\text{r},\Pi,j} \tag{5.57}$$

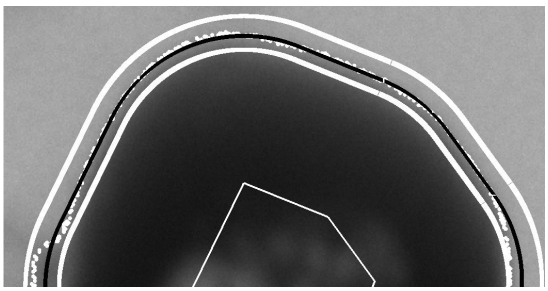

Figure 5.15.: Fitted shape (black line) with outer limiting shape according to $\max_i \delta(\mathbf{x}_i)^2$ and inner limiting shape according to $\min_i \delta(\mathbf{x}_i)$ (bold white lines)

where the opening angle α_j equals the angle between the normals of the adjacent edges. This averaging is consistent with cases where vertices of the identified circumscribing polytope disappear in the final kernel polytope. A suitable subset of data points that is used to fit the radius $\lambda_{\mathrm{r},\Pi,j}$ for a vertex j of the circumscribing polytope is obtained by finding the closest data point to the vertex j and then using 5% of the data points clockwise and 5% of the data points counterclockwise. Based on this subset of points, either the mean squares error $\delta_{\text{mean squared}}$ or the square height can be utilized. The optimal values $\lambda_{\mathrm{r},\Pi,j}$ are then obtained based on the golden section search [Kiefer, 1953]. In contrast to this local fitting approach, the second approach finds a radius $\lambda_{\mathrm{r},\Pi}$ by minimizing the mean squares error from equation 5.53. The optimal value $\lambda_{\mathrm{r},\Pi}$ is again obtained based on the golden section search [Kiefer, 1953]. For the sake of completeness, the square height is also considered for this global fitting approach. The four resulting approaches are denoted by local square height, global square height, local mean squares and global mean squares.

5.3.2 Simulation Case Study

Setup. To evaluate the four fitting approaches from the previous subsection, a simulation case study is presented in the following. It is constructed to emulate the crystal system and image quality according to subsection 5.3.3 where a real measurement setup is considered (see also figure 5.13). Four particles are constructed as they are shown in the top row of figure 5.16. The two aggregated particles in the right are considered because aggregated particles often occur in experimental data. Additionally, the local square height approach aimed to perform well for such cases which is not yet verified. These particles are altered according to the shape model such that a mean width based additive roundness from 0 to 1 is realized. This altered particles are then projected in 20 randomly chosen directions and a corresponding gray scale image is created for each projection direction like the ones shown in the middle of figure 5.16. The particle and image size is chosen such that the mean width of each primary particle is 200 pixels. In addition to that, the image is blurred by a Gaussian low pass filter and Gaussian white noise is added to resemble the image quality from experimental data (see subsection 5.3.3). The resulting images are shown in the last row of figure 5.16. Based on these images, the boundary coordinates of the projections are identified using a grayscale threshold and a Sobel filter for line detection. The threshold was fitted such that the resulting mean width of the shape identification algorithm matches the expected mean width of 200 pixels. These boundary points constitute the input to the shape identifica-

tion algorithm from the previous subsection.

Results. The computed mean width for each particle projection and for each of the four algorithms does not show a significant deviation from the expected value of 200 pixels. Deviations for the mean width would indicate that an appropriate threshold value is roundness dependent. The mean of the measured roundness values (μ^{B1}) was computed separately for each of the applied algorithms and separately for each initial particle to result in the graphs in figures 5.17 and 5.18. The x-axis represents the set roundness and the y-axis represents the measured roundness while the light blue line indicates the ideal behavior.

Discussion (general). Considering the blurred images already exhibiting certain roundness, it cannot be expected to be corrected by the identification algorithm. Therefore, the dashed light blue line, starting at 0.3 (measured roundness for set roundness 0) and linearly increasing to 1, provides a better reference for the performance of the algorithms. Another typical feature for the performance of all algorithms with respect to the convex particle example (left column of figure 5.16) is the truncation of the measured roundness towards 1 (left of figure 5.17). This behavior originates from the identified lines from which almost always small edges remain even for perfect circles so that a roundness of 1 cannot be measured. When more lines are allowed to be identified (*e.g.* by a larger value $\Delta_{\varphi,\mathrm{sep}}$, high roundness values would be truncated earlier).

Discussion (different approaches). The light black line represents the local mean squares approach and does not perform well for the ideal convex particle (left of figure 5.17). Especially its flatness spanning only the range from 0.3 to 0.6 makes this algorithm a comparably bad choice. The global square height approach, represented by the bold blue lines, performs best for the convex and bumped particle in the left of figure 5.16 (graphs in figure 5.17) while it shows a poor performance for the two-particle aggregate. The behavior of the local square height approach and the global mean squares is almost identical in figure 5.17 while the global mean squares performs better for the two-particle aggregate (third column in figure 5.16 and left in figure 5.18) since it spans a broader range of roundness values on the y-axis. The three-particle aggregate (fourth column in figure 5.16 and right in figure 5.18) does not yield any discrepancy information for the different approaches since all curves are equally flat.

Conclusion. In summary, the simulation experiment does not clearly identify a best algorithm and none of the algorithms already performs well for particle aggregates even though the local square height approach specifically focused on this objective. The main reason therefore is that the identified circumscribing polytope truncates many parts of the measured particle contour so that the subsequent fitting of the radii fails. Since the focused experimental case study in the next subsection does not consider aggregates, the algorithm was not improved on this behalf. Only the local mean squares approach yields significantly worse results so that it can be ignored in the following subsection.

5.3.3 Experimental Case Study

Motivation. Thanks to the group of Prof. Mazzotti (ETH Zürich, Stefan Schorsch in particular), a particle imaging setup [Schorsch et al., 2012] was available to test the image analysis procedure from subsection 5.3.1 with real measurement data. While it was aimed to measure the time evolution of particle size and roundness, the obtained

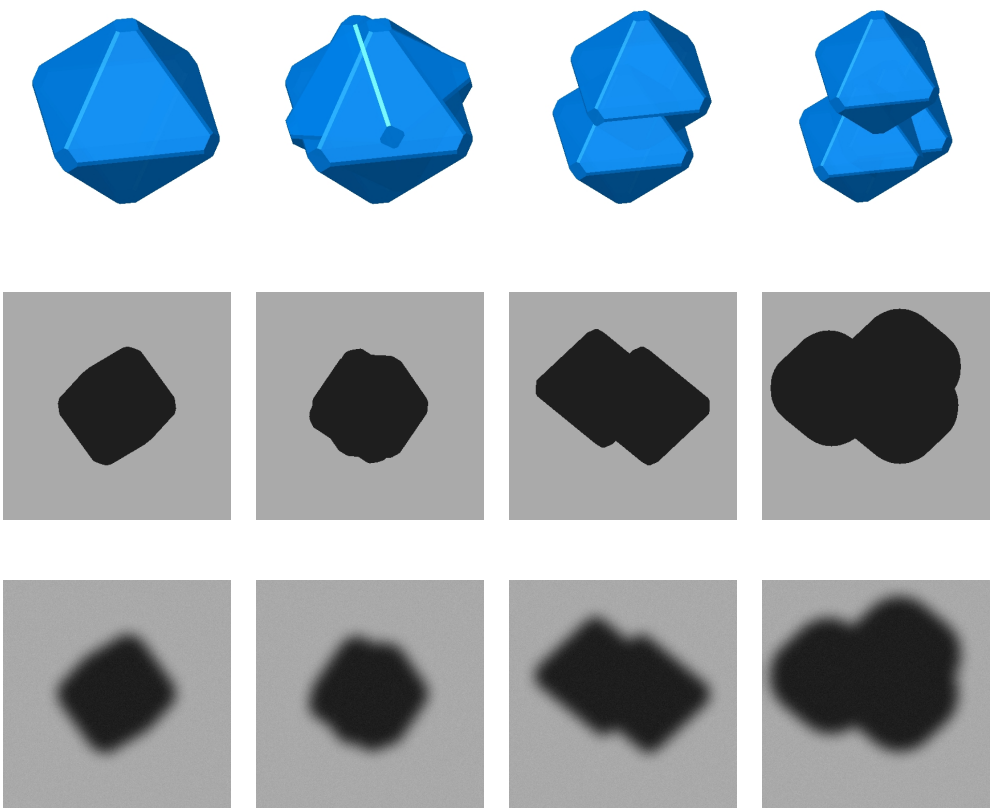

Figure 5.16.: Sample shapes for a simulation case study with the angular original particles in the top row, the projection images in the second row (roundness value in the outer left is: $\mu_{\mathrm{B}} = 0.1$, left: $\mu_{B1} = 0$, right: $\mu_{B1} = 0.05$, outer right: $\mu_{B1} = 0.6$) and the blurred images in the bottom row

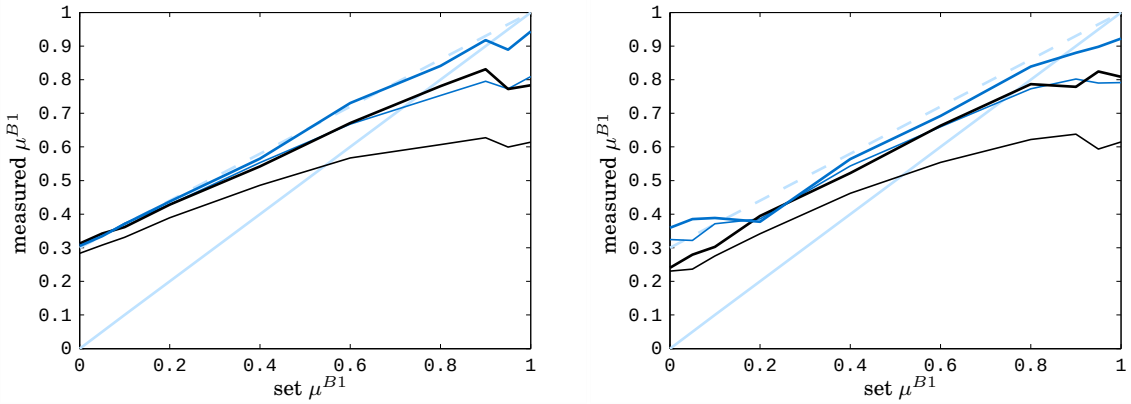

Figure 5.17.: Identified roundness μ_{B1} versus set roundness for the convex sample shape (first column in figure 5.16), left, and the bumpy shape (second column in figure 5.16), right, with bold lines for global approaches, light lines for local approaches, blue lines for square height and black lines for mean squares

results were not reproducible so that this subsection only evaluates the capabilities of the identification procedure for a few selected particle images to demonstrate that differences of roundness can properly be captured.

Experimental setup. The experiment design and evaluation of the results were mainly conducted in close collaboration during 4 weeks that I spent in Zürich while the experiments were executed solely by Stefan Schorsch. For the attrition experiment, aluminum potassium sulfate dodecahydrate (potash alum) in water was chosen. The crystal material was purchased from Sigma Aldrich with a purity $\geq 98\%$ and all used water is deionized by a MilliQ Advantage A10 system (Merck Chemicals). The crystals grow from an isometric Pa-3 space group with $a = 12.133\,\mathring{A}$ [Nyburg et al., 2000] where the faces $\{111\}$ are most prominent while they are accompanied by the $\{110\}$ and $\{100\}$ faces leading to an octahedral crystal shape as it is shown in the left of figure 5.19.

Seed particles were produced, preparing a saturated solution at $40\,^{\circ}\mathrm{C}$ which was cooled down rapidily to $30\,^{\circ}\mathrm{C}$. The resulting product material was collected and milled in a mortar to generate small particles. A second saturated solution at $30\,^{\circ}\mathrm{C}$ was prepared and the milled particles were added as seeds. The suspension was slowly cooled down to $20\,^{\circ}\mathrm{C}$ within 20 hours. The solid product was filtered and dried in open air. For the attrition experiments, saturated solutions of $1\,\mathrm{L}$ were prepared at $23\,^{\circ}\mathrm{C}$ and the reactor was kept at constant temperature by a controlled cooling jacket. Five grams of seed particles were added and the stirrer was set to $1800\,\mathrm{rpm}$. At every hour, the stirrer was slowed down to $250\,\mathrm{rpm}$ and crystals have been sampled and photographed for five minutes. The images were acquired by a stereoscopic imaging setup that has been previously introduced by [Schorsch et al., 2014]. Suspension is pumped from the crystallizer via a sampling loop through a sapphire glass flow-through cell which is illuminated by two flash lamps. Two orthogonally mounted 5 megapixel cameras simultaneously take pictures of crystals passing through the cell at a rate of $5\,\mathrm{Hz}$. The initial image analysis was performed in accordance with previous work comprising the following details [Schorsch et al., 2012, 2014]: grayscale threshold; contour extraction; matching of particles between both projections (*i.e.* in camera image 1 and camera image 2) with respect to common positions of objects in flow direction (z direction). The result of these operations are sets of contour coordinates in x/z direction and in y/z direction, respectively, that match the input data of the shape identification algorithm from subsection 5.3.1.

Results. Figure 5.20 displays column-wise 5 different samples of the initial seed particles, obtained by the stereographic setup. This means that projection directions in the first and second row (Π_1 and Π_2) are perpendicular to each other while both rows display the same particle. Likewise, figure 5.21 displays sample particles after 2 hours and figure 5.22 displays sample particles after 5 hours of stirring at $1800\,\mathrm{rpm}$. The bold white boundary for the images in all three figures illustrates the fitted outline of the projected particle shape according to equation 5.39 that is obtained by the global mean squares approach. The inner outlined shape indicates the kernel polytope. Additionally evaluated are the local and global square height approaches which typically compute similar fits as it is illustrated by figure 5.23 and table 5.2. Figure 5.23 contains three plots for the three different approaches where the measured mean width and roundness (μ^{B1}) for the particles in figures 5.20 (plusses), 5.21 (crosses) and 5.22 (circles) are indicated. Two matching projections are indicated by a connecting light blue line while specific particles are marked by the numbers from 1 to 5. To additionally emphasize the groups

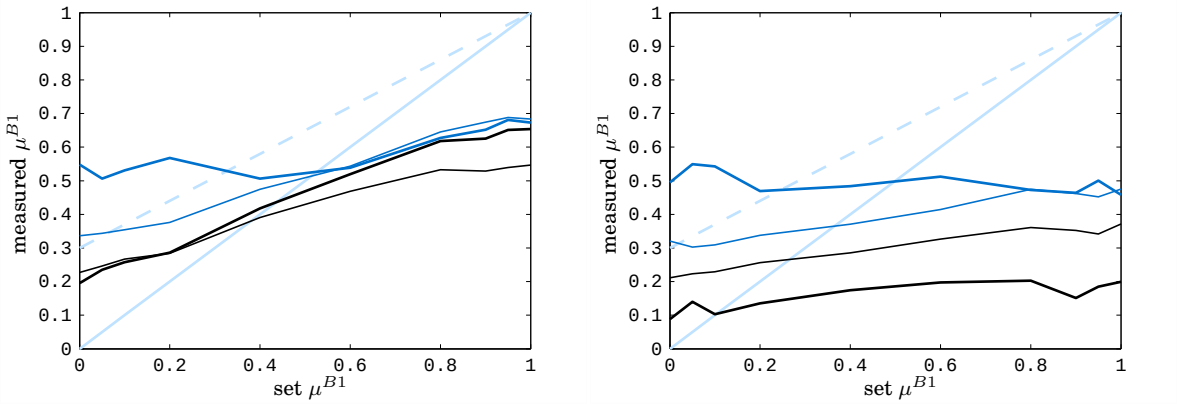

Figure 5.18.: Identified roundness μ_{B1} versus set roudness for the two-particle aggregate (third column in figure 5.16), left, and three-particle aggregate (fourth column in figure 5.16), right, with bold lines for global approaches, light lines for local approaches, blue lines for square height and black lines for mean squares

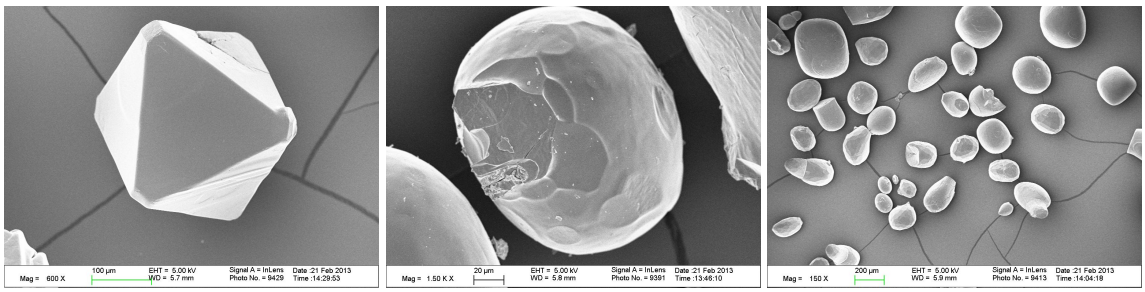

Figure 5.19.: SEM image of a seed crystal (left), a product crystal and an overview over product crystals.

of particles according to figures 5.20 to 5.22 or according to the corresponding sampling times (initial, 2 hours, 5 hours), respectively, the black line represents the convex hull of all data points that belong to the same sampling time. Table 5.2 lists the roundness values for all particle projections and algorithms and highlights values in bold that differ more than 0.1 from the result of another algorithm (for the same projection image).

Table 5.2.: Mean width based roundness values for three different approaches (square height is abbreviated by 'sq. h.' and mean squares by 'm. sq.') for the projected particle images in figures 5.20 (start 1 to 5), 5.21 (middle 1 to 5) and 5.22 (end 1 to 5); values are highlighted in bold when they differ more than 0.1 from the result of another algorithm for the same projection image

	Π_1			Π_2		
particle	local sq. h.	global sq. h.	global m. sq.	local sq. r.	global sq. r.	global m. sq.
start 1	0.28	0.29	0.31	0.37	0.46	0.37
start 2	0.30	0.36	0.30	0.30	0.25	**0.49**
start 3	0.36	0.37	0.30	0.29	**0.11**	0.27
start 4	0.35	0.32	0.34	0.33	0.41	0.35
start 5	0.19	0.26	0.20	0.20	**0.76**	0.20
middle 1	0.51	0.50	0.46	0.59	0.58	0.52
middle 2	0.54	0.54	0.62	0.57	0.57	0.51
middle 3	0.47	0.45	0.45	0.43	0.42	0.43
middle 4	**0.24**	**0.72**	**0.59**	0.62	0.58	0.60
middle 5	0.40	0.40	**0.53**	0.55	0.52	0.60
end 1	0.69	0.69	0.62	**0.77**	0.63	0.71
end 2	0.51	0.60	0.58	0.57	0.55	0.60
end 3	0.70	0.68	0.61	0.73	0.72	**0.60**
end 4	0.73	0.75	0.69	0.66	**0.60**	0.70
end 5	0.74	0.67	**0.63**	0.62	0.61	0.60

Discussion (circumscribing polytope). It occurs that the identified circumscribing polytope truncates the particle boundary like for the 4^{th} particle (projection Π_1) in figure 5.21. Given the parameterization of the lines according to the Hough transformation, this happens at polar angles where supporting hyperplanes would be supported by the boundary points of a well rounded edge. Corresponding bin values are typically very low (see equation 5.46). On the other hand, bumps create higher bin values for the same polar angle but at a significantly lower value for the polar distance of that line. These situations can be avoided by further narrowing the allowed polar distances in dependence of the polar angles (similar to equation 5.45) or by demanding that the boundary points that contribute to a bin value must be connected. However, such errors are rare and can be filtered well by evaluating the goodness of fit according to the profile height (equation 5.54).

Discussion (λ fitting approaches). Based on figure 5.23, all three algorithms properly distinguish angular from rounded particles. However, the global square height approach seems to be inferior to the other two approaches concerning the reliability. It often overestimates roundness (see table 5.2) when the particle contains outwards bumps (5^{th} particle, projection Π_2 in figure 5.20) or the particle contour is truncated (4^{th} particle, projection Π_1 in figure 5.21). On the other hand, inwards bumps can lead to an underestimated roundness (see table 5.2) like for the 3^{rd} example (projection Π_2) in figure 5.20. The inwards bump is hard to see and located in the lower right corner of that particle. Often, however, inwards and outwards bumps appear at the same time and with similar heights so that both effects are cancelled out and an appropriate mean contour line is found (4^{th} example, projection Π_2, in figure 5.20). In summary, the global profile height is not as reliable as the other two approaches while for these, no significant difference is found based on the analyzed data.

Conclusions. The developed image analysis and shape identification procedure successfully differentiates between angular and rounded particles in a reliable and quantitative manner (see also subsection 5.3.2). The best performing approaches for determining the radius of the added disk in the shape model are the global mean squares approach and the local square height approach. However, the local square height approach fails to evaluate the roundness for aggregates while it is slightly more complex if compared to the global mean squares approach. Therefore, the global mean squares approach should be preferred in applications.

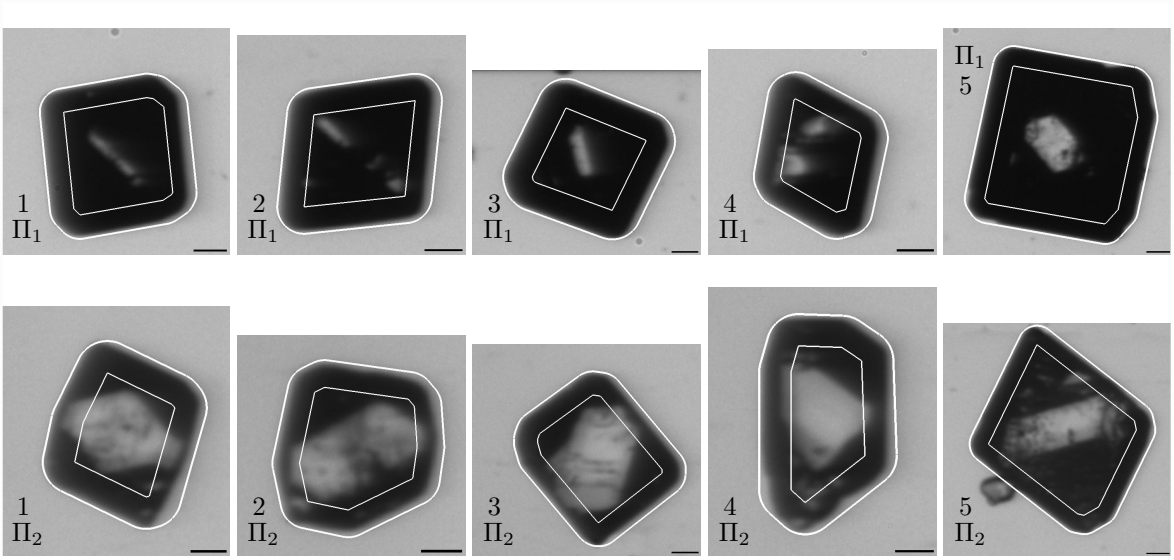

Figure 5.20.: Examples for the seed particles (columns 1 to 5) in two perpendicular projections (Π_1 and Π_2), fitted by the global mean squares approach

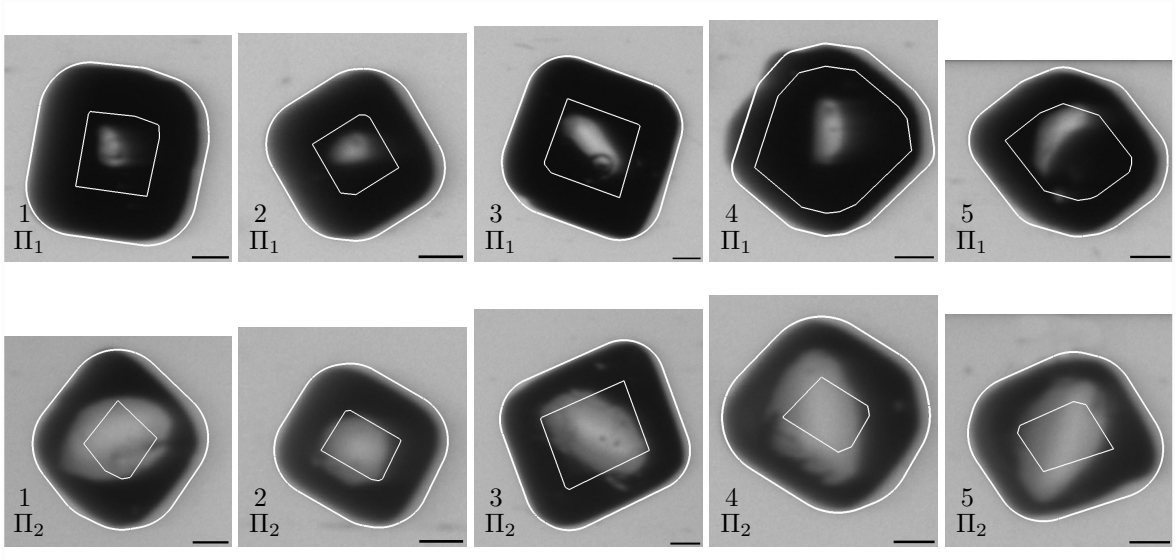

Figure 5.21.: Example particles after 2 hours of stirring (columns 1 to 5) in two perpendicular projections (Π_1 and Π_2), fitted by the global mean squares approach

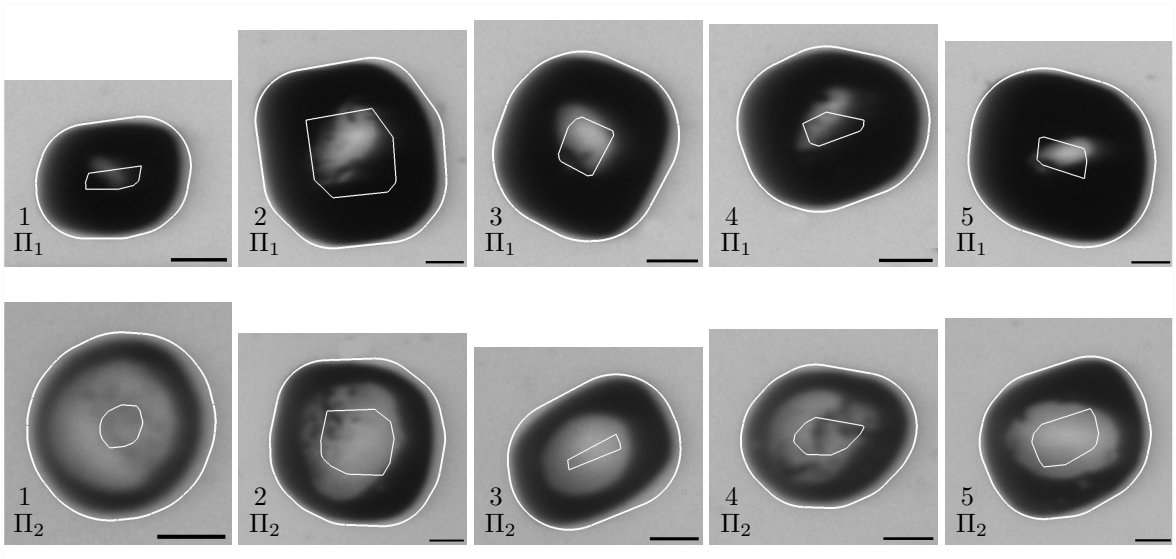

Figure 5.22.: Example particles after 5 hours of stirring (columns 1 to 5) in two perpendicular projections (Π_1 and Π_2), fitted by the global mean squares approach

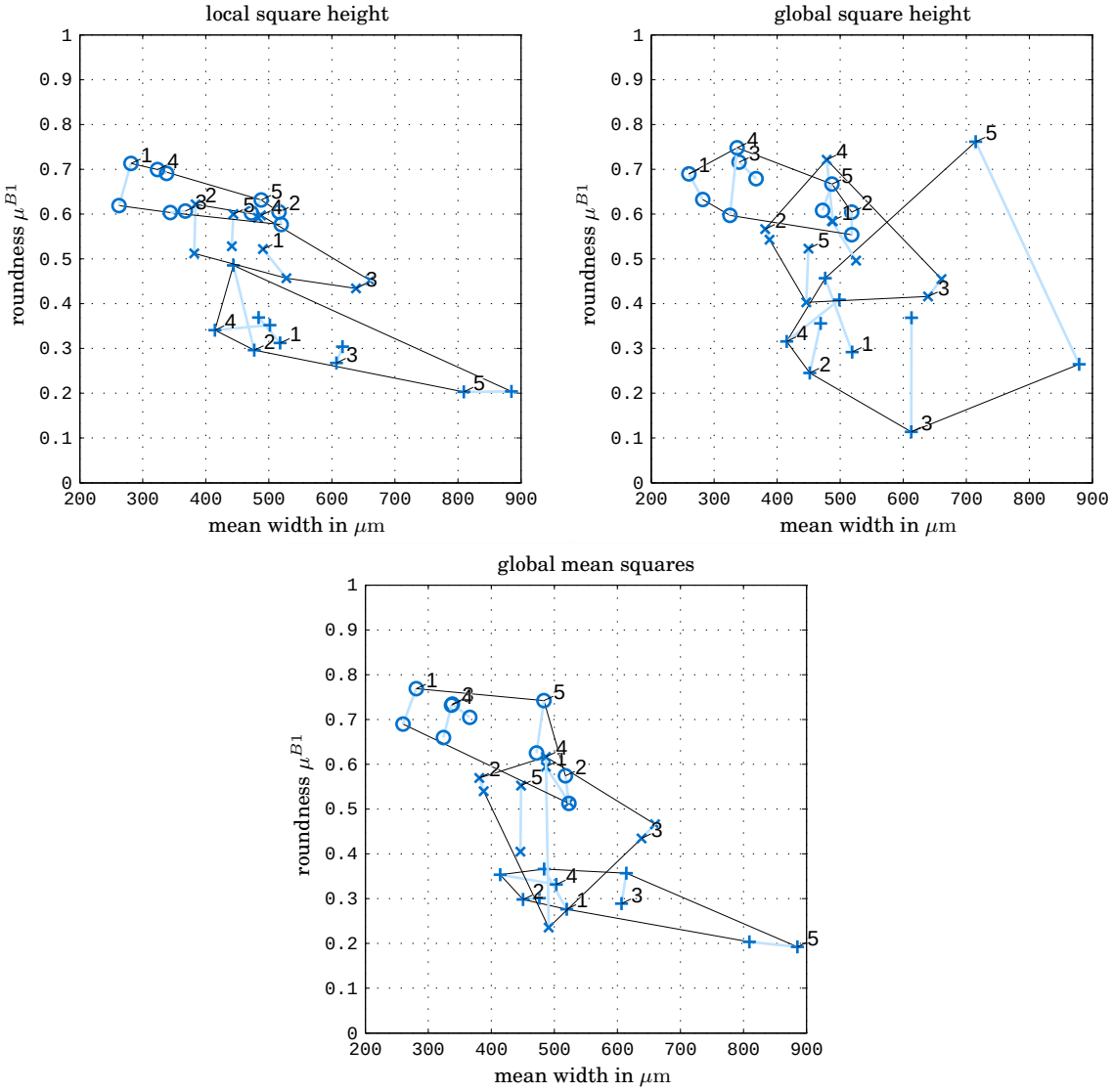

Figure 5.23.: Mean width and roundness μ^{B1} for the particles in figures 5.20 (plusses), 5.21 (crosses) and 5.22 (circles) according to three different approaches (the three plots); projections of the same particle are linked by light blue lines; data points of the same figure (or sampling time) are surrounded by a black line (convex hull).

6

Model Reduction and Shape Approximation

6.1 Introduction and Preliminaries

6.1.1 Introduction

Motivation. This chapter aims at a model reduction of the population balance that might include the necessity of crystal shape approximation. At a first thought, the term model reduction suggests a decreased dimension of the h_C-vectors that contain the independent shape parameters. Even though the accuracy and/or effort of crystallization processes simulations is not necessarily dependent on the dimension n_C (shown in chapter 4), such a reduction might be essential if the numerical scheme is changed from the approach that is followed in this work to, for example, a high resulution scheme with a regular grid. Another motivation can be posed by particle size and/or shape measurements. The desired properties of a particle population are often not directly available by measurements so that a reduction in the number of shape parameters might be useful for corresponding identification algorithms. On a second thought, the purpose of model reduction is manifested by a general relaxation of the complexity in applications. This complexity can arise from introducing more than one size coordinate but this is not a necessary consequence, as it is already outlined in the introduction of chapter 3. One example is measure calculation for which a case-by-case selection of coefficients is often required when $n_\mathrm{C} > 1$ applies. However, parallelepipeds and, in particular, cuboids can be modeled with 3 size coordinates for length, width and height. A corresponding $\mathcal{H}_\mathrm{C}$-representation has only one unified partition and, hence, only one set of coefficients for measure calculation even though $n_\mathrm{C} > 1$ holds. Section 6.3 presents a shape approximation that is constructed to achieve the outlined reduction in complexity. This shape approximation finally allows the reformulation of the population balance model into ordinary differential equations for the moments of the particle size/shape distribution. This result is particularly useful for real-time applications and model-based control.

Desirable starting point: $\mathcal{H}_\mathrm{C}$-representations. Assuming spherical particles with, for example, a volume equivalent diameter already represents a shape approximation since crystalline particles rarely possess a spherical shape. Slightly more adequate is the consideration of shape factors which imply the assumption of an arbitrary but con-

stant shape. More sophisticated are shape models that allow shape changes. One example of such models assumes a cuboid for arbitrary compact particles and uses length, width and height as shape parameters [Kempkes et al., 2010, Samad et al., 2011]. In contrast to these shape models, appropriate $\mathcal{H}_\mathrm{C}$-representations consider all facets of a crystal while assuming perfect symmetry [Borchert, 2012, Schorsch et al., 2014]. However, no work yet exists that derives a shape approximation based on such an ideal $\mathcal{H}_\mathrm{C}$-representation. Without such a connection, any shape approximation either uses a simple shape model for which measure computation remains simple (*e.g.* cuboids), or a suitably efficient concept for measure computation has to be derived from scratch. It is, hence, desirable to compute a shape approximation from a $\mathcal{H}_\mathrm{C}$-representation so that the insights for proper $\mathcal{H}_\mathrm{C}$-representations, including measure computation, transfer to the approximated shape model.

Outlook. In this section, the basic principles and the type of shape approximation is introduced based on a model reduction that does not require an approximated shape model. Section 6.2 then discusses a shape approximation that focuses on a reduction in the number of free variables while section 6.3 focuses on a reduction in the complexity for measure computations that allows a reformulation of the population balance into a set of ordinary differential equations without discretization.

Genuine research procedure. The presentation of this chapter does not follow the genuine research procedure. In chapter 3, proper $\mathcal{H}_\mathrm{C}$-representations were analyzed in great detail. The concept of valid $\mathbf{h}_\mathrm{C}$-vectors was introduced and an efficient way of measure calculation was derived from the decomposition of the validity cone into unified partitions. Because crystal symmetry ensures that the resulting $\mathcal{H}_\mathrm{C}$-representation is proper, this knowledge is sufficient for modeling the crystals in their ideal form. However, the forthcoming of modeling crystal shape was deemed to be blocked if the corresponding tight limitations of proper representations could not be overcome. One could either use proper $\mathcal{H}_\mathrm{C}$-representations, neglect shape changes completely or construct custom shape models for particular problems that imply only a simple geometry. At this point, shape approximations like to be presented in section 6.2 are the primary motivation to study improper $\mathcal{H}_\mathrm{C}$-representations and their peculiarities according to validity. Their analysis and the corresponding case studies, however, turned out that no significant benefit could be obtained by this theoretical analysis. Since this analysis is also comparably abstract, it is presented in appendix A.2 and not together with the successfull case studies that originally motivated it.

6.1.2 Model Reduction

Motivation. Let's assume a single crystal being modeled with several independently growing facet groups, *e.g.* $n_\mathrm{C} = 7$, but growing at a constant rate. Viewed in $\mathbf{h}_\mathrm{C}$-space, the trajectory of this particle follows a straight line parallel to the growth rate and displaced by the initial condition so that all $\mathbf{h}_\mathrm{C}$-vectors of the shape evolution are contained in some 2-dimensional linear subspace. Such a shape evolution could therefore be described perfectly by only 2 shape parameters. However, some facets might grow out at which point the trajectory hits the boundary of the validity cone. Since the $\mathbf{h}_\mathrm{C}$-vectors are required to remain valid, the growth rate is adopted and the trajectory bends to continue at the boundary of the validity cone. Such a trajectory cannot anymore be embedded in a 2-dimensional linear subspace.

The growth rate correction to maintain valid $\mathbf{h}_\mathrm{C}$-vectors was assumed obligatory for the dynamic shape evolution, discussed in section 3.2 and corresponding reasons are recalled in the following. First, validity guarantees an appropriate behavior when facets reappear during a simulation. Secondly, measure computation requires valid $\mathbf{h}_\mathrm{C}$-vectors since the coefficient sets $c^{\text{measure}}_{(\cdot),p}$ are only computed for unified partitions which are subdomains of the validity cone. These obstacles can be eluded. Facets that once disappeared during growth rarely reappear in common applications, rendering the first argument negligible. Additionally, valid $\mathbf{h}_\mathrm{C}$-vectors can be computed corresponding to originally invalid $\mathbf{h}_\mathrm{C}$-vectors to identify a matching unified partition and compute measures (see section 3.3).

In conclusion of the above statements, the dimension for the geometrical state space can be reduced to the degrees of freedom that are given by the initial condition and the growth rates if and only if disappeared facets never reappear. In such a case an alternative simulation procedure can be chosen. Instead of adopting the growth rate to maintain the validity of $\mathbf{h}_\mathrm{C}$-vectors, the growth rate remains unchanged and invalid $\mathbf{h}_\mathrm{C}$-vectors are mapped to valid $\mathbf{h}_\mathrm{C}$-vecors for the purpose of measure computation. This alternative simulation procedure provides the basis for all model reduction and shape approximation approaches in this chapter and the following demonstrates the achievable model reduction with two case studies. Additionally, this model reduction serves as a template for the shape approximation in section 6.2.

Single Particle Growth. In the first example, a crystal shape as illustrated in figure 6.1 is considered. It comprises the three opposing facets of a cuboid, three groups of facets that truncate the edges of the cuboid and one group of facets that truncates the vertices of the cuboid [Snyder et al., 2007]. The facet normal matrix is given by:

$$\mathbf{A}_\text{cuboid} = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix}^T \tag{6.1}$$

$$\mathbf{A}_\text{edge} = \frac{1}{\sqrt{2}} \begin{bmatrix} -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & 0 & 0 & 0 & 0 \\ -1 & 1 & -1 & 1 & 0 & 0 & 0 & 0 & -1 & -1 & 1 & 1 \\ 0 & 0 & 0 & 0 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \end{bmatrix}^T \tag{6.2}$$

$$\mathbf{A}_\text{vertex} = \frac{1}{\sqrt{3}} \begin{bmatrix} -1 & -1 & -1 & 1 & 1 & 1 & -1 & 1 \\ -1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 \\ -1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 \end{bmatrix}^T \tag{6.3}$$

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_\text{cuboid} \\ \mathbf{A}_\text{edge} \\ \mathbf{A}_\text{vertex} \end{bmatrix} \tag{6.4}$$

and the group mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}$ is chosen according to figure 6.1. Simulated is the growth of a single crystal for which all facet distances are set initially to $h_{\mathrm{C},i} = 10\,\mu\mathrm{m}$. The corresponding shape is shown in the left of figure 6.2. The crystal is grown for $3600\,\mathrm{s}$ with the rate $\mathbf{g}_\mathrm{C} = (0.2,\, 0.2,\, 0.4,\, 2,\, 2,\, 2,\, 6)^T \times 10^{-3}\,\frac{\mu\mathrm{m}}{\mathrm{s}}$ and some snapshots of the shape evolution are shown in figure 6.2. The shape change is almost complete after this period of time so that the right shape in that figure illustrates the growth shape $P(\mathbf{g}_\mathrm{C})$. This setup was simulated according to three different procedures: according

to the original procedure (see chapter 4) retaining valid $\mathbf{h}_\mathrm{C}$-vectors (original run); following the idea to retain the growth rates and fix invalid $\mathbf{h}_\mathrm{C}$-vectors only for measure computation (no-growth-limitation run); according to the reduction of the model which is given in the following (model reduction run).

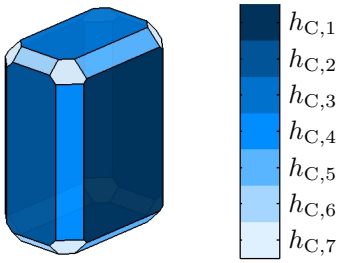

Figure 6.1.: Facet groups for a cuboid with truncated edges and vertices

Model reduction (single particle). For the particular case of constant growth rates, the particle trajectory for the no-growth-limitation run is embedded in a linear subspace that is spanned by the initial condition and the growth rate. Therefore, a new space of 2-dimensional vectors $\mathbf{h}_\mathrm{A}$ can be constructed with the mapping:

$$\mathbf{h}_\mathrm{C} = \mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\mathbf{h}_\mathrm{A}. \tag{6.5}$$

$$\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}} = \begin{bmatrix} \frac{\mathbf{h}_\mathrm{C}(t=0)}{\|\mathbf{h}_\mathrm{C}(t=0)\|} & \frac{\mathbf{g}_\mathrm{C}}{\|\mathbf{g}_\mathrm{C}\|} \end{bmatrix}. \tag{6.6}$$

Note that the new $\mathbf{h}_\mathrm{A}$ vectors also form a $\mathcal{H}_\mathrm{C}$-representation with a new group mapping matrix:

$$\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}} = \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}. \tag{6.7}$$

The growth model can be transferred into this new shape representation using the pseudo inverse matrix $\mathbf{M}^+_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$:

$$\frac{\mathrm{d}\mathbf{h}_\mathrm{A}}{\mathrm{d}t} = \mathbf{g}_\mathrm{A} = \mathbf{M}^+_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\mathbf{g}_\mathrm{C}. \tag{6.8}$$

$$\mathbf{h}_\mathrm{A}(t=0) = \mathbf{M}^+_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\mathbf{h}_\mathrm{C}(t=0) \tag{6.9}$$

The simulation of this reduced model was performed and the resulting vectors $\mathbf{h}_\mathrm{A}$ are lifted to the original $\mathbf{h}_\mathrm{C}$-space by equation 6.5 for the comparison to the original and no-growth-limitation run.

Notation. While the index A is chosen arbitrarily for this subsection, it is consistent with the shape approximation in section 6.2 and reads 'approximated'. Since the $\mathbf{h}_\mathrm{A}$-vectors also form a $\mathcal{H}_\mathrm{C}$-representation, existing symbols must be adopted. Every appearance of 'C' in a symbol is substituted by 'A', reading 'approximating' even though the polytopes $P(\mathbf{h}_\mathrm{A})$ are not yet a shape approximation to the polytopes $P(\mathbf{h}_\mathrm{C})$ in the scope of this subsection. If 'C' does not appear in the symbol, '/A' is appended to the sub- or superscript, reading 'in approximated space'. An overview of these new symbols is given in column 3 of table 6.1 where column 4 already summarizes symbols for a third $\mathcal{H}_\mathrm{C}$-representation that is used in section 6.3 and appendix A.2.

Results (single particle). Figure 6.3 demonstrates the growth of the particle over time based on its facet distances and mean width. At $t \approx 660\,\mathrm{s}$ the facets that truncate the cuboid vertices disappear and at between $t \approx 2460\,\mathrm{s}$ and $t \approx 2640\,\mathrm{s}$, the facets that truncated the cuboid edges also disappear. These events are visible by the slope changes and also affect the evolution of the mean width. The left of figure 6.4 displays the relative errors between the original and the no-growth-limitation run. Two peaks are visible at the times where facets disappear. They originate from the stiffness handling that was introduced for the growth rate corrections and, therefore, represent an error stemming from the original run and not from the no-growth-limitation run. For the same reason, the no-growth-limitation run is typically used as basis for comparisons in this chapter. The right of figure 6.4 displays the relative error between the no-growth-limitation run and the model reduction run. The relative errors are in the range of machine precision which verifies the model reduction from originally 7 shape parameters to 2 shape parameters without any loss of information.

Dissolution of a Particle Distribution. In the second example, the generic cuboid crystal from above is reused. The initial condition comprises a shape distribution with 3 degrees of freedom and the particles are dissolved rather than grown. The initial particles have a mean shape according to the left example in figure 6.5 which uses $h_{\mathrm{C},1} = 500\,\mu\mathrm{m}$, $h_{\mathrm{C},2} = 1000\,\mu\mathrm{m}$ and $h_{\mathrm{C},3} = 1500\,\mu\mathrm{m}$. To simulate a shape distribution, 100 sample points were selected according to a normal distribution with the standard deviations $\mathrm{std}\left(h_{\mathrm{C},1}\right) = 25\,\mu\mathrm{m}$, $\mathrm{std}\left(h_{\mathrm{C},2}\right) = 50\,\mu\mathrm{m}$ and $\mathrm{std}\left(h_{\mathrm{C},3}\right) = 75\,\mu\mathrm{m}$. The remaining shape coordinates are computed such that the corresponding facets have disappeared while the $\mathbf{h}_{\mathrm{C}}$-vectors are valid. In consequence, all $\mathbf{h}_{\mathrm{C}}$-vectors are embedded in some 3-dimensional linear subspace. The dissolution rate $\mathbf{g}_{\mathrm{C}} = (-2.5,\, -2.5,\, -5,\, -25,\, -25,\, -25,\, -75)^T \times 10^{-2}\,\frac{\mu\mathrm{m}}{\mathrm{s}}$ was selected and some snapshots of the shape evolution are shown in figure 6.5.

Model reduction (particle population). The same simulation runs were executed as for the previous example. In this case, the projection matrix $\mathbf{M}_{\mathbf{h}_{\mathrm{A}} \mapsto \mathbf{h}_{\mathrm{C}}}$ is computed from the matrix:

$$\begin{bmatrix} \frac{\mathbf{g}_{\mathrm{C}}}{\|\mathbf{g}_{\mathrm{C}}\|} & \frac{\mathbf{h}_{\mathrm{C},1}(t=0)}{\left\|\mathbf{h}_{\mathrm{C},1}(t=0)\right\|} & \cdots & \frac{\mathbf{h}_{\mathrm{C},100}(t=0)}{\left\|\mathbf{h}_{\mathrm{C},100}(t=0)\right\|} \end{bmatrix}$$

where the vectors $\mathbf{h}_{\mathrm{C},i}(t = 0)$ are the sample points of the particle distribution. The mapping matrix $\mathbf{M}_{\mathbf{h}_{\mathrm{A}} \mapsto \mathbf{h}_{\mathrm{C}}}$ is obtained by a singular value decomposition so that its range matches the range of the matrix above. The resulting $\mathbf{h}_{\mathrm{A}}$-vectors are 4-dimensional.

Results (particle population). Figure 6.6 shows in the left the time evolution of the mean facet distances and mean width over time and, additionally, the total number of remaining sample points in the right. The particle distribution vanishes between $t = 2000\,\mathrm{s}$ and $t = 2500\,\mathrm{s}$. The cuboid facets disappear completely while the facets that truncate the cuboid initially appear but vanish again later in time. The relative errors in figure 6.7 are computed based on the mean facet distances. Again, the comparison between the no-growth-limitation and original run shows errors at time points were facets disappear. The relative error between the no-growth-limitation run and model reduction run has relative errors in the range of 10^{-11} up to $t = 2100\,\mathrm{s}$ were sample points already disappear. The comparably large errors thereafter are caused by different numbers of remaining sample points during the dissolution process. These differing numbers are, however, not caused by errors in the $\mathbf{h}_{\mathrm{C}}$-vectors. They originate from the

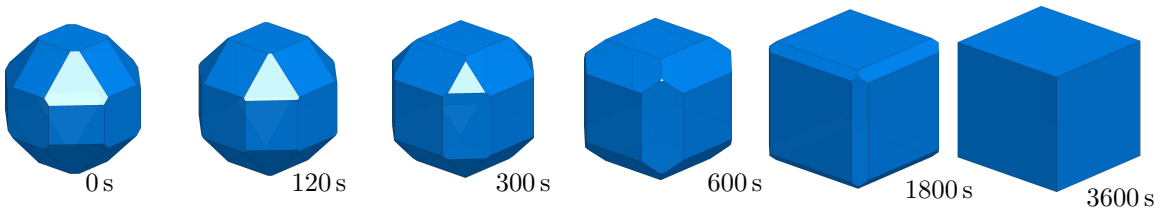

Figure 6.2.: Shape evolution of a growing crystal (crystal size is normalized)

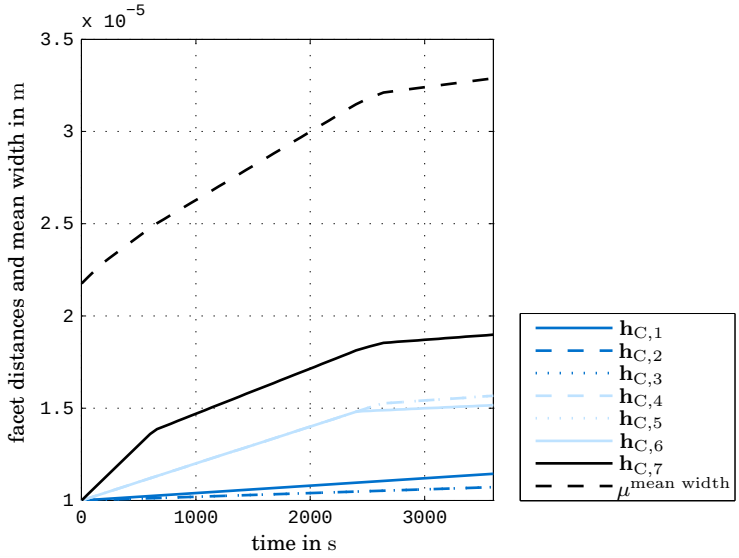

Figure 6.3.: Evolution of facet distances and mean width for the growth of a truncated cuboid.

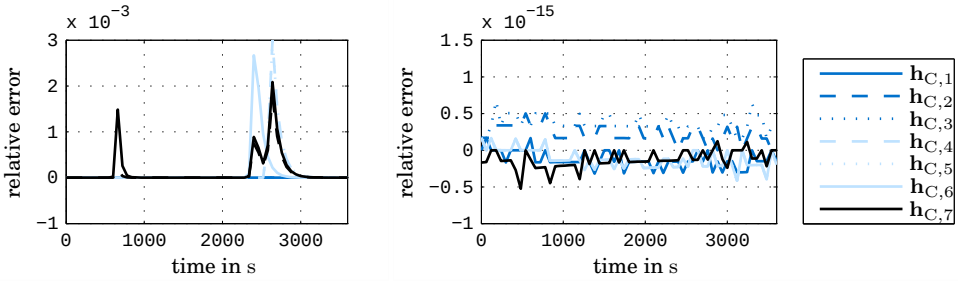

Figure 6.4.: Relative errors between the no-growth-limitation and original run (left) and between the no-growth-limitation and model reduction run (right): growth of a truncated cuboid.

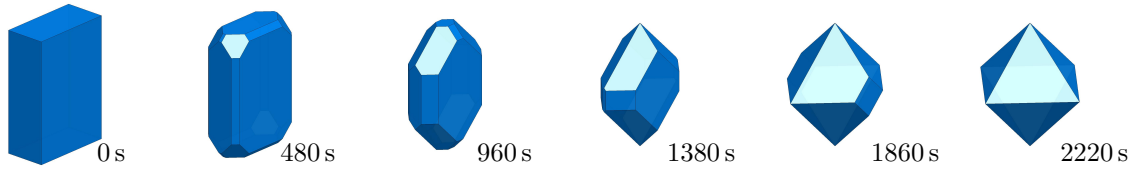

Figure 6.5.: Shape evolution of a dissolving crystal (crystal size is normalized)

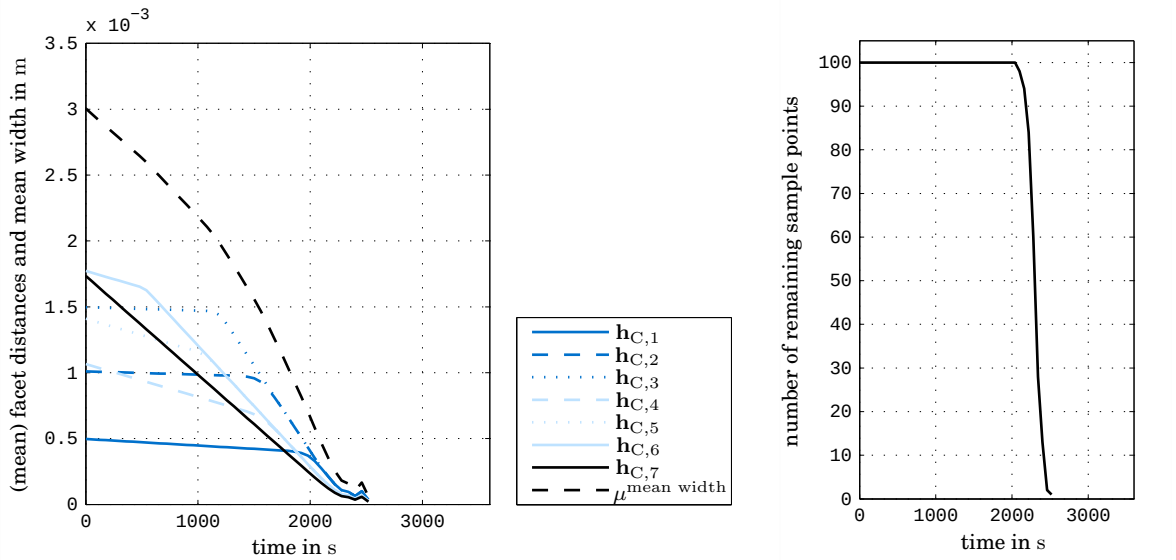

Figure 6.6.: Evolution of the mean facet distances and mean width (left) and the number of sample points (right) for the dissolution of a truncated cuboid.

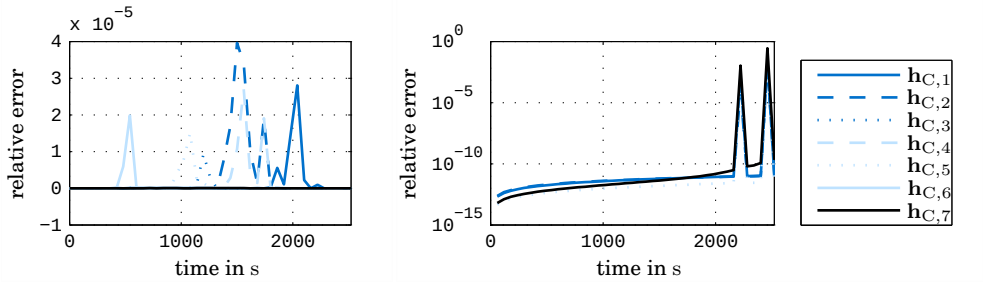

Figure 6.7.: Relative errors between the no-growth-limitation and original run (left) and between the no-growth-limitation and model reduction run (right): dissolution of a truncated cuboid.

method that is used to determine disappearing sample points. A sample point disappears when the distance of a $\mathbf{h}_\mathrm{C}$-vector from the empty polytope cone is smaller than some tolerance. The empty polytope cone is mapped to the $\mathbf{h}_\mathrm{A}$-space using the facet normal matrix:

$$\mathbf{A}_{\mathrm{FV}(0)/\mathrm{A}} = \mathbf{A}_{\mathrm{FV}(0)}\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}.$$

Since the distance of a $\mathbf{h}_\mathrm{C}$-vector from the non-empty polytope cone differs from the distance of the corresponding $\mathbf{h}_\mathrm{A}$-vector from the mapped non-empty polytope cone, the sample points are removed at different points in time. Therefore, it can be concluded that the model reduction to a 4-dimensional subspace, again, characterizes the originally 7-dimensional problem without any loss of information.

Summary. The geometrical state space can be reduced in dimension whenever the combined degrees of freedom from the initial condition and growth rate are lower than n_C. While the above examples used linear subspaces, the idea also extends to nonlinear subspaces that might be found for shape dependent growth. Required components for such a model reduction are:

1. a mapping function that describes the new n_A-dimensional space, embedded in the original $\mathbf{h}_\mathrm{C}$-space: $f_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}} : \mathbb{R}^{n_\mathrm{A}} \mapsto \mathbb{R}^{n_\mathrm{C}}$
2. and a reverse mapping function: $f_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{A}} : \mathbb{R}^{n_\mathrm{C}} \mapsto \mathbb{R}^{n_\mathrm{A}}$.

In the presented linear cases, these mappings are:

$$f_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}(\mathbf{h}_\mathrm{A}) = \mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\mathbf{h}_\mathrm{A} \tag{6.10}$$

and

$$f_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{A}}(\mathbf{h}_\mathrm{C}) = \mathbf{M}^{+}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\mathbf{h}_\mathrm{C}. \tag{6.11}$$

The first mapping is required to utilize the insights of the original (proper) $\mathcal{H}_\mathrm{C}$-representation which includes, for example, measure computation and the empty polytope cone. The second mapping is required to transfer the initial condition and growth rate to the reduced space of $\mathbf{h}_\mathrm{A}$-vecors.

Table 6.1.: Overview of symbols for $\mathcal{H}_\mathrm{C}$-representations including new symbols in the context of model reduction or shape approximation

Symbols	Description	New Symbols: approximation	New Symbols: embedding
$\mathcal{H}$, $\mathcal{H}_\mathrm{C}$	Descriptor of the representation		
$\mathbf{A}$, $\mathbf{h}$, n_H	matrix of facet normals, facet distance vector for $\mathcal{H}$-representation and number of facets		$\tilde{\mathbf{A}}$, $\tilde{\mathbf{h}}$, $\tilde{n}_\mathrm{H}$ (extended, only)
$\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}$, $\mathbf{h}_\mathrm{C}$	mapping matrix and reduced vector of facet distances	$\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}}$, $\mathbf{h}_\mathrm{A}$	$\mathbf{M}_{\mathbf{h}_\mathrm{E}\mapsto\mathbf{h}}$, $\mathbf{h}_\mathrm{E}$
	mapping matrices for the conversion to and from the $\mathbf{h}_\mathrm{C}$-space	$\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$, $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{A}}$	$\mathbf{M}_{\mathbf{h}_\mathrm{E}\mapsto\mathbf{h}_\mathrm{C}}$, $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}$ and $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\tilde{\mathbf{h}}}$ (extended, only)
	mapping matrices between embedding and approximation	$\mathbf{M}_{\mathbf{h}_\mathrm{E}\mapsto\mathbf{h}_\mathrm{A}}$, $\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{E}}$	
$\mathcal{C}_\mathrm{V}$, $\mathbf{A}_\mathrm{V}$	validity cone and corresponding facet normal matrix	$\mathcal{C}_{\mathrm{V/A}}$, $\mathbf{A}_{\mathrm{V/A}}$	$\mathcal{C}_{\mathrm{V/E}}$, $\mathbf{A}_{\mathrm{V/E}}$
$\mathcal{C}_{\mathrm{FV}(i)}$, $\mathbf{A}_{\mathrm{FV}(i)}$	facet validity cones and corresponding facet normal matrices	$\mathcal{C}_{\mathrm{FV}(i)/\mathrm{A}}$, $\mathbf{A}_{\mathrm{FV}(i)/\mathrm{A}}$	$\mathcal{C}_{\mathrm{FV}(i)/\mathrm{E}}$, $\mathbf{A}_{\mathrm{FV}(i)/\mathrm{E}}$
$\mathcal{C}_{\mathrm{U}(i)}$, $\mathbf{A}_{\mathrm{U}(i)}$	unified paritions and corresponding facet normal matrices	$\mathcal{C}_{\mathrm{U}(i)/\mathrm{A}}$, $\mathbf{A}_{\mathrm{U}(i)/\mathrm{A}}$	$\mathcal{C}_{\mathrm{U}(i)/\mathrm{E}}$, $\mathbf{A}_{\mathrm{U}(i)/\mathrm{E}}$
S_i, $\mathbf{h}_{\mathrm{C},S_i}$	structuring elements and corresponding reduced vector of facet normals	$\mathbf{h}_{\mathrm{A},S_i}$	$\mathbf{h}_{\mathrm{E},S_i}$
$\mathbf{P}_\mathrm{V}$, $\mathbf{A}_{\mathbf{P}_\mathrm{V}}$	validity projection for a $\mathbf{h}_\mathrm{C}$-vector and facet normal matrix for the applicable region of the projection	$\mathbf{P}_{\mathrm{V/A}}$, $\mathbf{A}_{\mathbf{P}_{\mathrm{V/A}}}$	$\mathbf{P}_{\mathrm{V/E}}$, $\mathbf{A}_{\mathbf{P}_{\mathrm{V/E}}}$
$c^{\mathrm{measure}}_{p,(i,j,k)}$	coefficients for measure calculation	$c^{\mathrm{measure/A}}_{p,(i,j,k)}$	$c^{\mathrm{measure/E}}_{p,(i,j,k)}$

6.2 Reduction of Dimension

6.2.1 Problem Statement

Motivation. The model reduction demonstrated in the previous section is applicable to only a very limited set of problems. More typical problems comprise variations that span the entire $\mathbf{h}_\mathrm{C}$-space where some degrees of freedom are important, like the change in size and some major shape changes, and some degrees of freedom are comparably negligible, like variations in the initial condition or minor shape changes. For these cases, the shape approximation that is introduced in this subsection aims to cover the important degrees of freedom while ignoring the minor ones.

General Principles. To obtain a shape approximation that couples to a known $\mathcal{H}_\mathrm{C}$-representation, the same mapping functions $f_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$ and $f_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{A}}$ are required like for the model reduction from the previous section. The range of the function $f_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$ describes the lower dimensional subspace embedded in the original $\mathbf{h}_\mathrm{C}$-space and, hence, the range of shapes that are covered by the new representation. The function $f_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{A}}$ describes the approximation of $\mathbf{h}_\mathrm{C}$-vectors that are not in the aforementioned range of the new representation. This implies that each value from the function $f_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{A}}$ implicitly or explicitly results from an optimization problem:

$$f_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{A}}\left(\mathbf{h}_\mathrm{C}\right) = \arg\min_{\mathbf{h}_\mathrm{A}} f\left(f_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\left(\mathbf{h}_\mathrm{A}\right), \mathbf{h}_\mathrm{C}\right) \tag{6.12}$$

where f describes some error between the original vector $\mathbf{h}_\mathrm{C}$ and its approximation by the vector $\mathbf{h}_\mathrm{A}$. To ensure that the error f is zero when a vector $\mathbf{h}_\mathrm{C}$ in the range of $f_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$ is chosen, the function $f_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{A}}$ must fulfill a consistency condition:

$$\mathbf{h}_\mathrm{A} = f_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{A}}\left(f_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\left(\mathbf{h}_\mathrm{A}\right)\right). \tag{6.13}$$

Limitations. In the scope of this chapter, only shape approximations are discussed that use linear functions $f_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$. The range of such a shape approximation describes a linear subspace embedded in the space of the original $\mathcal{H}_\mathrm{C}$-representation. Figure 6.8 illustrates such a situation with the validity cone of the original $\mathcal{H}_\mathrm{C}$-representation in light blue and the linear subspace of a possible approximation in dark blue. The resulting shape representation is then also a $\mathcal{H}_\mathrm{C}$-representation. Corresponding adopted symbols are introduced in the previous subsection and summarized in column 3 of table 6.1.

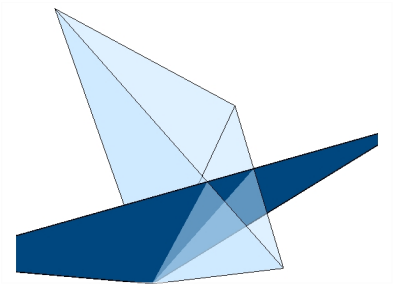

Figure 6.8.: Validity cone of a $\mathcal{H}_\mathrm{C}$-representation in light blue and the embedded $\mathbf{h}_\mathrm{A}$-space of an approximated $\mathcal{H}_\mathrm{C}$-representation in dark blue

A second limitation to this approach results from the general optimization problem from equation 6.12 which is fixed to:

$$f_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{A}}\left(\mathbf{h}_\mathrm{C}\right) \quad = \quad \underset{\mathbf{h}_\mathrm{A}}{\arg\min} \left\|\mathbf{h}_\mathrm{C} - f_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\left(\mathbf{h}_\mathrm{A}\right)\right\| \tag{6.14}$$

which is solved by the pseudo inverse matrix $\mathbf{M}^{+}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$:

$$f_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{A}}\left(\mathbf{h}_\mathrm{C}\right) = \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{A}}\mathbf{h}_\mathrm{C} = \mathbf{M}^{+}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\mathbf{h}_\mathrm{C} \tag{6.15}$$

and minimizes the Euclidian distance between the original $\mathbf{h}_\mathrm{C}$-vector and its approximation: $\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{A}}\mathbf{h}_\mathrm{C}$.

Problem statement. The shape approximation in this section works analogous to the model reduction from subsection 6.1.2. It aims to reduce the dimension of a given $\mathcal{H}_\mathrm{C}$-representation by finding an optimal mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$. The corresponding optimization problem:

$$\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}} \quad = \quad \underset{\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}}{\arg\min}\ f_\mathrm{A}\left(\mathbf{H}_\mathrm{C},\, \mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\right). \tag{6.16}$$

is based on a set of sample vectors $\mathbf{h}_{\mathrm{C},i}$ that are arranged in the matrix:

$$\mathbf{H}_\mathrm{C} = \left[\ \mathbf{h}_{\mathrm{C},i} \quad \cdots \quad \mathbf{h}_{\mathrm{C},n_\mathrm{sample}}\ \right]. \tag{6.17}$$

It fixes the mapping function $f_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$ while the mapping function $f_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{A}}$ is defined by equation 6.15.

Principal component analysis. The work of Wang and Ma [2009] might already be regarded as a shape approximation according to the problem statement above. Based on simulation data for potash alum crystals in $\mathcal{H}_\mathrm{C}$-representation ($n_\mathrm{C} = 3$), they generated virtual measurement data in $\mathcal{H}$-representation by adding 10% Gaussian noise. This virtual measurement data was then reduced by principal component analysis (PCA) to 3 principal coordinates, the dimension of the original simulation data, and the growth rates were identified. This implies that instead of obtaining a reduced shape representation from an original $\mathcal{H}_\mathrm{C}$-representation and transferring the known growth rates, they aimed to identify a $\mathcal{H}_\mathrm{C}$-representation and corresponding growth rates from measurement data. Their work is, hence, different from the aim in this section.

The PCA approach nevertheless provides a first objective function for the general approximation problem from equation 6.16:

$$f_{\mathrm{A,\,PCA}}\left(\mathbf{H}_\mathrm{C},\, \mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\right) \quad = \quad \sum_i \left(\mathbf{h}_{\mathrm{C},i} - \mathbf{P}_\mathrm{A}\mathbf{h}_{\mathrm{C},i}\right)^T \left(\mathbf{h}_{\mathrm{C},i} - \mathbf{P}_\mathrm{A}\mathbf{h}_{\mathrm{C},i}\right) \tag{6.18}$$

$$\mathbf{P}_\mathrm{A} \quad = \quad \mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\mathbf{M}^{+}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}. \tag{6.19}$$

Its solution can be provided by a singular value decomposition of the matrix [Jolliffe, 2002]:

$$\mathbf{H}_\mathrm{C} = \mathbf{U}_\mathrm{PCA}\mathbf{\Sigma}_\mathrm{PCA}\mathbf{V}^T_\mathrm{PCA} \tag{6.20}$$

where the first n_A columns of $\mathbf{U}_\mathrm{PCA}$ determine the mapping matrix:

$$\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}} = [\mathbf{u}_{\mathrm{PCA},1},\, \ldots,\, \mathbf{u}_{\mathrm{PCA},n_\mathrm{A}}]. \tag{6.21}$$

This approach minimizes the (empirical) variance of the $\mathbf{h}_\mathrm{C}$-vector components while assuming the mean: $\mathbf{h}_\mathrm{C} = \mathbf{0}$. Therefore, the first PCA component $h_{\mathrm{A},1}$ reflects mostly the particle size. As it is typical for PCA, the obtained solution is sensible to a scaling of the coordinate axes. Additionally, since the objective is based on the squared absolute error, shape variations for large particles influence the result more than shape variations of very small particles. Despite of these drawbacks, the PCA approach is useful because its solution can be computed easily.

Measure based objectives. The particle volume is often the most important particle property since it determines the product yield and is required to determine the supersaturation and growth rates. Other measures like the surface area might also be important when they determine the product quality. The PCA approach does not consider corresponding errors and, hence, cannot produce a shape approximation that minimizes relevant simulation errors. As an example, figure 6.9 illustrates the setup of a urea crystal [Salvalaglio et al., 2013]. Given this hypothetical shape, it is evident that variations in the $h_{\mathrm{C},3}$-coordinate influence the particle volume much less than variations in the $h_{\mathrm{C},1}$-coordinate. Therefore, the following objectives are proposed:

$$f_{\mathrm{A,abs}^2}\left(\mathbf{H}_\mathrm{C},\, \mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\right) = \sum_{i,j} \gamma_{\mu,j} \left(\mu_j(\mathbf{h}_{\mathrm{C},i}) - \mu_j(\mathbf{P}_\mathrm{A}\mathbf{h}_{\mathrm{C},i})\right)^2 \tag{6.22}$$

$$f_{\mathrm{A,rel\,max}}\left(\mathbf{H}_\mathrm{C},\, \mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\right) = \sum_{j} \gamma_{\mu,j} \left(\max_i \left| \frac{\mu_j(\mathbf{h}_{\mathrm{C},i}) - \mu_j(\mathbf{P}_\mathrm{A}\mathbf{h}_{\mathrm{C},i})}{\mu_j(\mathbf{h}_{\mathrm{C},i})} \right| \right) \tag{6.23}$$

with μ_j being arbitrary particle measures and $\gamma_{\mu,j}$ being corresponding weights. The first objective $f_{\mathrm{A,abs}^2}$ is a generalization of the PCA objective $f_{\mathrm{A,\,PCA}}$ which is obtained for $\mu_j = h_{\mathrm{C},j,i}$ and $\gamma_{\mu,j} = 1$. The second objective $f_{\mathrm{A,rel\,max}}$ aims to put a minimal upper limit to all considered relative errors.

Sample point generation. The set of sample points $\mathbf{h}_{\mathrm{C},i}$ could be generated manually based on experience or, if available, based on measurement data [Singh et al., 2012, Schorsch et al., 2014, Kovačević et al., 2014]. Another alternative is to generate the sample points by a suitable simulation. In the case of the urea example (figure 6.9), a hypothetical crystal growth is simulated and used as an example throughout the next subsection. The shape illustrated in figure 6.9 determines the initial condition and uses the dimensionless facet distances $\mathbf{h}_\mathrm{C} = (0.5,\, 1,\, 1.3)^T$. The growth is simulated in dimensionless time from $t = 0$ till $t = 2$ while the rate $\mathbf{g}_\mathrm{C} = (0.1,\, 0.5,\, 0.75)^T$ applies for $0 \leq t \leq 1$ and the rate $\mathbf{g}_\mathrm{C} = (0.1,\, 0.75,\, 1.5)^T$ for $t > 0$. Figure 6.10 shows some snapshots of the shape evolution while figure 6.11 shows the evolution of the coordinates $h_{\mathrm{C},i}$ and mean width. At $t \approx 1.55$, the facet group that is described by $h_{\mathrm{C},3}$ grows out. This is visible by the dashed line which represents valid $\mathbf{h}_\mathrm{C}$-vectors while the solid lines represent the evolution of the $\mathbf{h}_\mathrm{C}$-vectors without any adoption of the growth rates. The $\mathbf{h}_\mathrm{C}$-vectors that are sampled at regular time steps $t \in \{0,\, 0.01,\, 0.02,\, \ldots,\, 2\}$ are generated as sample points $\mathbf{h}_{\mathrm{C},i}$.

Complexity of the optimization problem. The optimization problem that is posed by equation 6.16 is difficult. The number of free variables that describe the matrix $\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$ is very high with $n_\mathrm{C} \times n_\mathrm{A}$. Additionally, no reasonable lower or upper boundaries are known for the components of the matrix $\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$. On the other hand, the optimization problem has a multiplicity of equivalent solutions. The matrix $\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$

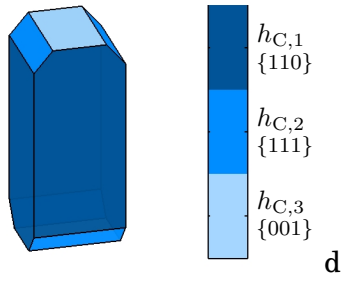

Figure 6.9.: Crystal shape example (urea, Salvalaglio et al. [2013]).

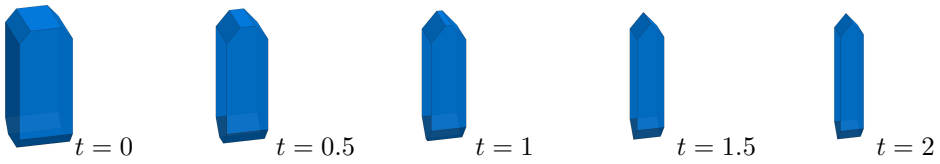

Figure 6.10.: Snapshots of the shape evolution (plotted shapes are shown with normalized size)

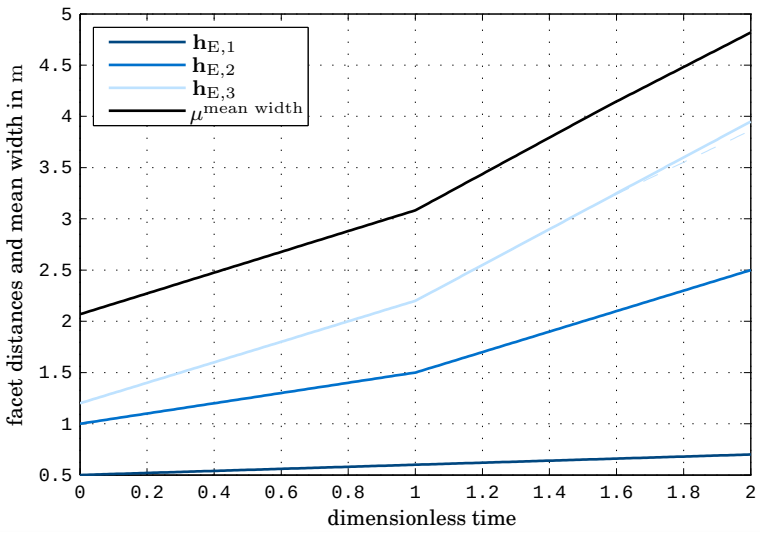

Figure 6.11.: Shape evolution and evolution of mean with for the growing urea crystal (dashed lines show the evolution of valid vectors $\mathbf{h}_E$)

contains the base vectors of the linear subspace in which the approximated shape vectors $\mathbf{P}_\mathrm{A}\mathbf{h}_\mathrm{C}$ are situated. Hence, the approximation result is identical when two matrices $\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$ describe the same linear subspace. It would, hence, be desirable to reduce the total number of free variables by corresponding constraints. First, the length of the base vectors can be fixed to 1 and orthogonality can be claimed for all base vectors which results in the constraint:

$$\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}^T\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}} = \mathbf{I}_{n_\mathrm{A}}. \tag{6.24}$$

With this constraint, the base vectors describe an orthonormal coordinate system. Another constraint would consider rotations or reflections of the $\mathbf{h}_\mathrm{A}$-vectors in their own subspace which can be represented by the coordinate transformation:

$$\mathbf{h}'_\mathrm{A} = \mathbf{O}_{n_\mathrm{A}}\mathbf{h}_\mathrm{A} \tag{6.25}$$

that uses any matrix for which:

$$\mathbf{O}_{n_\mathrm{A}}^T\mathbf{O}_{n_\mathrm{A}} = \mathbf{I}_{n_\mathrm{A}} \tag{6.26}$$

holds. Applying this transformation to the projection matrix $\mathbf{P}_\mathrm{A}$ (equation 6.19) yields:

$$\mathbf{P}_\mathrm{A} = \left(\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\mathbf{O}_{n_\mathrm{A}}\right)\left(\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\mathbf{O}_{n_\mathrm{A}}\right)^+ \tag{6.27}$$

$$= \mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\mathbf{O}_{n_\mathrm{A}}\mathbf{O}_{n_\mathrm{A}}^T\mathbf{M}^+_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}} \tag{6.28}$$

so that the objective function is invariant with respect to such coordinate transformations:

$$f_\mathrm{A}\left(\mathbf{H}_\mathrm{C},\, \mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\right) = f_\mathrm{A}\left(\mathbf{H}_\mathrm{C},\, \mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}\mathbf{O}_{n_\mathrm{A}}\right). \tag{6.29}$$

The constraint for the orthonormality of the matrix $\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$ fixes $\frac{1}{2}\left(n_\mathrm{A}^2 + n_\mathrm{A}\right)$ degrees of freedom, constituted by the upper triangular matrix elements of equation 6.24. The transformation of the $\mathbf{h}_\mathrm{A}$-space according to $\mathbf{O}_{n_\mathrm{A}}$ implies $\frac{1}{2}\left(n_\mathrm{A}^2 - n_\mathrm{A}\right)$ degrees of freedom. This is the case since equation 6.26 does not describe constraints to the matrix $\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$ but constraints to the modeled degrees of freedom, indicated by equation 6.29. Unfortunately, no direct constraints to the matrix $\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$ can be provided for these degrees of freedom. In total, n_A^2 constraints apply and the optimization problem could be reduced to $n_\mathrm{A}\left(n_\mathrm{C} - n_\mathrm{A}\right)$ free variables. However, since this reduction of the free variables is not straightforward, the solution of the optimization problem (equation 6.16) is treated separately in the next subsection.

6.2.2 Solution of the Problem Statement

Grassmann manifolds. Given the optimization problem from equation 6.16, the discussion in the preceding subsection already concluded that the objective function f_A only depends on the linear subspace that is spanned by the columns of the matrix $\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$. In mathematical literature, the set of n_A-dimensional linear subspaces that are embedded in an n_C-dimensional space is known as the Grassmann manifold $\mathcal{G}_{n_\mathrm{C},n_\mathrm{A}}$ [Milnor and Stasheff, 1974, Edelman et al., 1998, Absil et al., 2008]. For example, $\mathcal{G}_{3,2}$ represents the set of all planes in $\mathbb{R}^3$ that go through $\mathbf{0}$. A Grassmann manifold is a non-linear, homogeneous $n_\mathrm{A}(n_\mathrm{C} - n_\mathrm{A})$-dimensional space that can be embedded in the space $\mathbb{R}^{n_\mathrm{C}\times n_\mathrm{A}}$. The dimension of the Grassmann manifold matches the degrees of freedom that were already introduced in the preceding subsection while the embedding space represents the components of the matrix $\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$. In general, manifolds behave locally like an Euclidean space so that distances and angles (in particular orthogonality)

can be determined. While the global properties of manifolds are more complex, these properties are not relevant.

Procedure. In the following, the theory that is mostly taken from Edelman et al. [1998] is utilized to determine a reduced set of $n_\mathrm{A}(n_\mathrm{C} - n_\mathrm{A})$ free variables including corresponding lower and upper bounds for the original optimization problem in equation 6.16. Based on the new coordinate system, the optimization problem is reformulated and solved by a general purpose global optimization algorithm, available in MATLAB®. To describe the theoretical concepts, a temporarily simplified notation is introduced. The $n_\mathrm{C} \times n_\mathrm{A}$ matrix $\mathbf{Y}$, analogous to the matrix $\mathbf{M}_{\mathbf{h}_\mathrm{A} \mapsto \mathbf{h}_\mathrm{C}}$, represents a point on the Grassmann manifold $\mathcal{G}_{n_\mathrm{C}, n_\mathrm{A}}$. The term 'represents' is important at this stage since the Grassmann manifold describes linear subspaces while the matrix $\mathbf{Y}$ describes a specific set of base vectors for this linear subspace.

Tangent space. In this first step, the tangent space for a point $\mathbf{Y}$ on the Grassmann manifold is introduced that discloses the local Euclidean space and, hence, the concept of directions on the manifold. To visualize the principle of a tangent space, imagine a surface in $\mathbb{R}^3$. This surface is a 2-dimensional manifold. For an arbitrary but fixed point on this manifold, many tangents exist that can all be chosen from a plane touching the surface in the given point. This plane is called the tangent space for general manifolds. For a Grassmann manifold and a fixed point $\mathbf{Y}_0$, the tangent space is given by the columns of [Edelman et al., 1998]:

$$\mathbf{Y}_\perp \mathbf{B} \tag{6.30}$$

where $\mathbf{Y}_\perp$ is a $n_\mathrm{C} \times (n_\mathrm{C} - n_\mathrm{A})$ matrix that contains any set of orthonormal basis vectors for the null space of $\mathbf{Y}_0$:

$$\mathbf{Y}_\perp^T \mathbf{Y}_\perp = \mathbf{I}_{n_\mathrm{C} - n_\mathrm{A}} \tag{6.31}$$

$$\mathbf{Y}_0^T \mathbf{Y}_\perp = \mathbf{0} \tag{6.32}$$

and $\mathbf{B}$ being any $(n_\mathrm{C} - n_\mathrm{A}) \times n_\mathrm{A}$ matrix. Equation 6.30 describes $n_\mathrm{C} \times n_\mathrm{A}$ matrices and, hence, points in the embedding space while the matrix $\mathbf{B}$ determines the degrees of freedom for the tangent space. In general, the tangent space of a manifold always has the same dimension as the manifold itself. In fact, the tangents identify directions on the manifold which might become more apparent in the following.

Geodesics. Given two points of the manifold, the shortest path on this manifold lies on a geodesic. This defines a geodesic and since it is a path, the dimension of a geodesic is always 1. Additionally, each geodesic through $\mathbf{Y}_0$ crosses $\mathbf{Y}_0$ parallel to a specific tangent direction. This implies two viewpoints. First, geodesics provide the principles to move on a manifold, given a starting point $\mathbf{Y}_0$ and a specific direction from the tangent space. Secondly, since any point $\mathbf{Y}$ on the manifold is connected to the arbitrary but fixed point $\mathbf{Y}_0$, the point $\mathbf{Y}$ can be described by the point $\mathbf{Y}_0$, a direction from the tangent space of $\mathbf{Y}_0$ and a distance t. According to Edelman et al. [1998], geodesics through the fixed point $\mathbf{Y}_0$ on a Grassmann manifolds are described by[1]:

$$\mathbf{Y}(t) = \mathbf{Y}_0 \mathbf{V} \cos(\mathbf{\Sigma} t) \mathbf{V}^T + \mathbf{U} \sin(\mathbf{\Sigma} t) \mathbf{V}^T \tag{6.33}$$

[1]The terms $\cos(\mathbf{\Sigma} t)$ and $\sin(\mathbf{\Sigma} t)$ apply the trigonometric functions element-wise.

where $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ is the compact singular value decomposition of $\mathbf{Y}_\perp\mathbf{B}$ and t is the free coordinate of the path, equivalent to a signed distance. For a compact singular value decomposition, the matrix $\mathbf{\Sigma}$ has the dimension $n_\mathrm{R} \times n_\mathrm{R}$ with $n_\mathrm{R} = \operatorname{rank}(\mathbf{Y}_\perp\mathbf{B})$. Corresponding to the singular value decomposition, the parameter t is redundant to a scaling of $\mathbf{B}$ so that $t = 1$ can be chosen to obtain:

$$\mathbf{Y}(\mathbf{B}) = \mathbf{Y}_0\mathbf{V}\cos(\mathbf{\Sigma})\,\mathbf{V}^T + \mathbf{U}\sin(\mathbf{\Sigma})\,\mathbf{V}^T. \tag{6.34}$$

This equation describes any point $\mathbf{Y}$ on the Grassmann manifold by an arbitrary origin $\mathbf{Y}_0$ and the choice of the matrix $\mathbf{B}$. Hence, only $n_\mathrm{A}(n_\mathrm{C} - n_\mathrm{A})$ variables are used to determine the matrix $\mathbf{Y}$.

Constraints on $\mathbf{B}$. Edelman et al. [1998, equation 2.67] provides a relation between the singular values σ_i in the diagonal elements of the matrix $\mathbf{\Sigma}$ and the components of $\mathbf{B}$:

$$\sqrt{\sum_i \sigma_i^2} = \sqrt{\operatorname{tr}\left((\mathbf{Y}_\perp\mathbf{B})^T(\mathbf{Y}_\perp\mathbf{B})\right)} = \sqrt{\operatorname{tr}(\mathbf{B}^T\mathbf{B})} = \sqrt{\sum_{i,j} b_{i,j}^2}. \tag{6.35}$$

This relation and equation 6.34 is utilized in the following to determine lower and upper bounds for the components of the matrix $\mathbf{B}$. Apparently, the geodesic describes identical matrices $\mathbf{Y}$ for the singular values $\sigma_i = \sigma_i \pm 2k\pi$, with $k \in \{0, 1, \ldots\}$. Since $\mathbf{Y}$ and $-\mathbf{Y}$ both describe the same subspace, even the singular values $\sigma_i = \sigma_i \pm k\pi$ are equivalent. Additionally, singular values are always positive so that a negative direction $\sigma_i < 0$ would be represented by $\sigma_i > 0$ but with a sign change in a column of $\mathbf{U}$ or $\mathbf{V}$. In summary, $\sigma_i \leq \frac{\pi}{2}$ can be demanded for the singular values of $\mathbf{Y}_\perp\mathbf{B}$ while still every point on the manifold is covered. The singular values are equivalent to so called principal angles so that $\sigma_i \leq \frac{\pi}{2}$ can alternatively be concluded by the work of Edelman et al. [1998, page 337] and Hamm [2008]. From this constraint on the singular values and equation 6.35, it follows:

$$\sum_{i,j} b_{i,j}^2 = \sum_i \sigma_i^2 \quad \leq \quad \min(n_\mathrm{A},\, n_\mathrm{C} - n_\mathrm{A}) \left(\frac{\pi}{2}\right)^2 \tag{6.36}$$

which is also formulated by Wong [1967, theorem 8b]. This can be interpreted geometrically when all free variables $b_{i,j}$ are aligned in a vector. This vector must then be contained in the $n_\mathrm{A}(n_\mathrm{C} - n_\mathrm{A})$-dimensional unit hypersphere scaled to the radius:

$$\sqrt{\min(n_\mathrm{A},\, n_\mathrm{C} - n_\mathrm{A})}\,\frac{\pi}{2}$$

to avoid ambiguous points on the Grassmann manifold. In addition to this tight limitations, the variables $b_{i,j}$ can alternatively be bounded by:

$$-\sqrt{\min(n_\mathrm{A},\, n_\mathrm{C} - n_\mathrm{A})}\,\frac{\pi}{2} \leq b_{i,j} \leq \sqrt{\min(n_\mathrm{A},\, n_\mathrm{C} - n_\mathrm{A})}\,\frac{\pi}{2}. \tag{6.37}$$

Reformulation of the optimization problem. The optimization problem can now be reformulated to:

$$\mathbf{B} \quad = \quad \underset{\mathbf{B}\in\mathbb{R}^{(n_\mathrm{C}-n_\mathrm{A})\times n_\mathrm{A}}}{\arg\min} \quad f_\mathrm{A}\left(\mathbf{H}_\mathrm{C},\, \mathbf{Y}(\mathbf{B})\right). \tag{6.38}$$

with upper and lower bounds for the components of $\mathbf{B}$ given in equation 6.37. For the fixed point $\mathbf{Y}_0$ on the Grassmann manifold, the solution of the PCA approximation problem is used. The result of the optimization problem are the computed elements of $\mathbf{B}$ while the desired mapping matrix is then given by equation 6.34:

$$\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}} = \mathbf{Y}(\mathbf{B}). \tag{6.39}$$

However, in following case studies the matrix $\mathbf{B}$ or the mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$ are typically not reported since their values do not map easily to geometric properties. Instead, the results are analysed by relevant errors of the simulation and, if applicable, by the display of approximated shape samples.

Scaling. While the above formulation already resolves the issues of the original optimization problem according to the number of free variables and appropriate boundaries, a last detail must be considered to guarantee an efficient optimization. The sample points typically cover only a small subdomain of the $\mathbf{h}_\mathrm{C}$-space so that many linear subspaces will not intersect with the convex hull of the data points. Such linear subspaces necessarily yield poor objective values and should, hence, be avoided. A coordinate transformation according to the principal component analysis or singular value decomposition (equation 123) can resolve this issue. The rotation according to the matrix $\mathbf{U}_\mathrm{PCA}^T$ aligns the largest variance in the data, which typically correlates with the particle size, to the first coordinate axis and the subsequent coordinate axes cover decreasing variance in the data. A scaling of these new coordinate axes according to $\mathbf{\Sigma}_\mathrm{PCA}^{-1}$ then ensures equal variance for all axes. For the data points in this new coordinate system, the convex hull is not anymore narrow and most of the linear subspaces intersect with the data points. To use the overall coordinate transformation in the optimization procedure, the linear subspaces are modeled in the scaled space. A new reference point:

$$\hat{\mathbf{Y}}_0 \;=\; \mathbf{\Sigma}_\mathrm{PCA}^{-1}\mathbf{U}_\mathrm{PCA}^T\mathbf{Y}_0 = \begin{bmatrix} \mathbf{I}_{n_\mathrm{C}} \\ \mathbf{0} \end{bmatrix} \tag{6.40}$$

on the Grassmann manifold is used and base vectors of the linear subspace in the original data space are recovered by:

$$\mathbf{Y}(\mathbf{B}) = \mathbf{U}_\mathrm{PCA}\mathbf{\Sigma}_\mathrm{PCA}\hat{\mathbf{Y}}(\mathbf{B}). \tag{6.41}$$

This scaling implies that solutions are significantly different from the PCA solution are reduced to small areas in the $b_{i,j}$-space which is illustrated in the following examples.

Figure 6.12 (left) illustrates the PCA objective function $f_{\mathrm{A,\,PCA}}$ (equation 6.18) for the sample points from the urea simulation (see page 124) for the unscaled $\mathbf{h}_\mathrm{C}$-space with $b_{1,1}$ and $b_{2,1}$ between -2π and 2π. The black circle indicates the boundaries to the free variables in $\mathbf{B}$ according to equation 6.36 and the objective function forms only a narrow valley inside of these bounds. Outside of these bounds, the periodicity of the objective function is visible. Similar observations can be made for the left of figure 6.13 which illustrates the objective $f_{\mathrm{A,\,rel\,max}}$ (equation 6.23) for the volume measure ($\mu_1 = \mu^{\mathrm{volume}}$, $\gamma_{\mu,1} = 1$). However, it seems to contain two crossing valleys while one of the valleys is aligned identically to the valley of the PCA objective. The plots in the middle of both figures show the same objective functions based on the scaling above. Figure 6.12 (PCA objective) demonstrates that most of the regions in the $b_{i,j}$-space cover reasonable good

values for the objective function while poor objective values are obtained only at the boundary according to equation 6.36. For the objective $f_{\mathrm{A,\,rel\,max}}$ in figure 6.13, however, the valley that corresponds to the PCA valley is deformed to the broad spot in the middle while the second valley is visible by a wide arc close to the boundary of the free variables. This demonstrates two general issues. First, most of the $b_{i,j}$ values are attracted by the local optimum near the center of the $b_{i,j}$-space. Secondly, typical gradient based optimization algorithms will perform badly for the narrow curved valley in the top which seems to contain the global optimum.

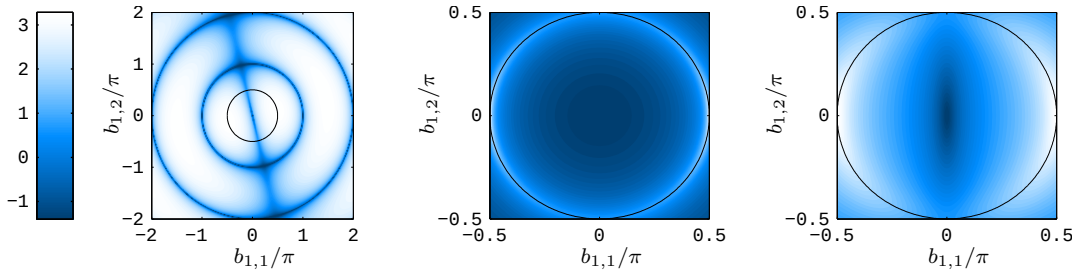

Figure 6.12.: Plotted objective function $f_{\mathrm{A,\,PCA}}$ (equation 6.18) for the sample points according to the urea growth simulation (see page 124) for an unscaled $\mathbf{h}_{\mathrm{C}}$-space (left), the scaling according to equations 6.40 and 6.41 (middle) and the scaling according to equation 6.42 and 6.43 (right); dark spots indicate local optima

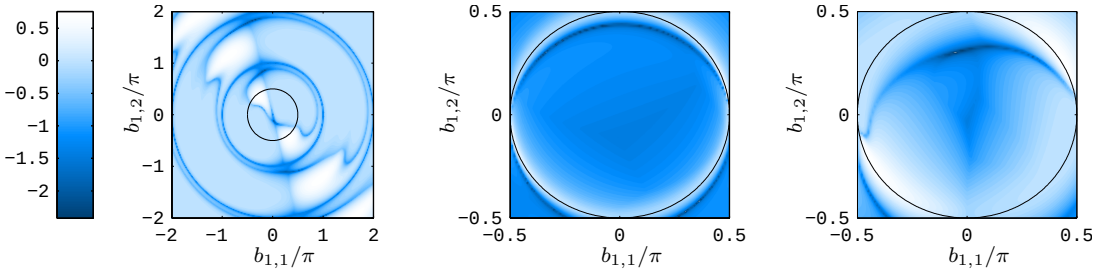

Figure 6.13.: Plotted objective function $f_{\mathrm{A,\,rel\,max}}$ (equation 6.23 with $\mu_1 = \mu^{\mathrm{volume}}$, $\gamma_{\mu,1} = 1$) for the sample points according to the urea growth simulation (see page 124) for an unscaled $\mathbf{h}_{\mathrm{C}}$-space (left), the scaling according to equations 6.40 and 6.41 (middle) and the scaling according to equation 6.42 and 6.43 (right); dark spots indicate local optima

Based on the discussion above, the scaling can be relaxed by using the square root of the singular values in the diagonal elements of the new matrix $\mathbf{\Sigma}^{0.5}_{\mathrm{PCA}}$. This new scaling is applied analogue to the approach above:

$$\hat{\mathbf{Y}}_0 = \left(\mathbf{\Sigma}^{0.5}_{\mathrm{PCA}}\right)^{-1} \mathbf{U}^T_{\mathrm{PCA}} \mathbf{Y}_0 \tag{6.42}$$

$$\mathbf{Y}(\mathbf{B}) = \mathbf{U}_{\mathrm{PCA}} \mathbf{\Sigma}^{0.5}_{\mathrm{PCA}} \hat{\mathbf{Y}}(\mathbf{B}). \tag{6.43}$$

The right of figure 6.12 demonstrates that for this new scaling, regions of the $b_{i,j}$-space with good values of the objective function are present in similar portions compared to

regions of the $b_{i,j}$-space with poor values. This indicates that any solution is well represented in $b_{i,j}$-space while the narrow domain of the data points is properly emphasized. Figure 6.13 for the objective $f_{\mathrm{A,\,rel\,max}}$ also gives evidence for a reasonable scaling. The crossing of the two valleys remains vaguely visible, the valleys containing the optima are only slightly bent and a much larger portion of the $b_{i,j}$-space is attracted by the valley that is assumed to contain the global optimum.

Numerical solution. The optimization problem typically contains multiple local minima which is also illustrated by figure 6.13 (most apparent in the right). It is solved by an optimization algorithm readily available in MATLAB®. The `GlobalSearch` algorithm [Ugray et al., 2007] provides a strategy for an efficient selection of starting points for a local optimizer for which sequential quadratic programming via `fmincon` is used [Fletcher and Powell, 1963, Goldfarb, 1970]. The setup of the local solver is unproblematic. It considers only tolerances for the objective value and free variables. The parameters for the `GlobalSearch` algorithm, however, determine whether the global optimum is found. In the first stage of the algorithm an adjustable amount of points is sampled from the rectangular region that is specified by equation 6.37. In the second stage, an adjustable amount of local optimization runs is executed with the starting points selected from stage one. Starting points that yield poor objective values are neglected as well as starting points that are close to already known local optima. Therefore, new sample points are also generated during stage two of the algorithm to refill the points from stage one.

It must be noted, additionally, that dedicated optimization algorithms are available that operate directly on the Grassmann manifold [Edelman et al., 1998, Absil et al., 2008, Adragni et al., 2012]. They require an analytic formulation of the partial derivatives of the objective function in dependence of the components of the matrix $\mathbf{Y}$. While a significant decrease of the computational cost can be expected with analytical gradients, this approach does not provide any additional insight for this work. Additionally, the derivation and implementation of these gradients is an error prone task. Therefore, the optimization problem is solved as discussed earlier.

Results for the urea growth example. Figure 6.14 shows the absolute relative errors for the volume (black), surface area (blue) and mean width (light blue) and for different objective functions. Only the errors according to the projection of the original sample points (equation 6.19) represented by the bold dashed lines are considered for now. The left plot represents the PCA solution ($f_{\mathrm{A,\,PCA}}$, equation 6.18) which is the basis of comparison since it can be computed easily by singular value decomposition. This solution was also used to verify the implementation of the global optimization problem with the objective $f_{\mathrm{A,\,PCA}}$. For the middle plot, the objective function $f_{\mathrm{A,\,abs^2}}$ (equation 6.22) is used and the right plot uses $f_{\mathrm{A,\,rel\,max}}$ (equation 6.23). It is assumed that the volume measure shall be approximated best so that $\mu_1 = \mu^{\mathrm{volume}}$ and $\gamma_{\mu,1} = 1$ apply. For the PCA solution, the highest absolute relative error is $4,7\%$ at $t = 0$ for the volume. The solution for the objective function $f_{\mathrm{A,\,abs^2}}$ shows a significantly increased accuracy for the volume. The maximum absolute relative error is 0.3% at $t = 0$. In contrast to that, the accuracy of the surface area has improved only slightly while the mean width is approximated worse. The results for the objective function $f_{\mathrm{A,\,rel\,max}}$ (right) differ only slightly from the results for $f_{\mathrm{A,\,abs^2}}$ (middle). The large relative error at $t = 0$ is positive (not visible) and decreased for the $f_{\mathrm{A,\,rel\,max}}$-solution. In return, the negative relative

error at $t \approx 0.8$ increased slightly. It can be concluded that the differences between the objective function $f_{\mathrm{A,\,rel\,max}}$ and $f_{\mathrm{A,\,abs^2}}$ are well reflected by the different optimal solutions and the objective function $f_{\mathrm{A,\,rel\,max}}$ successfully evens out negative and positive relative errors to obtain a maximum absolute value of the relative error of 0.18% for the volume measure. This error is less than 4% of the original error while the computation of the approximation itself took less than a minute [Desktop Computer, 2009]. Computation times for the actual simulation are negleglible.

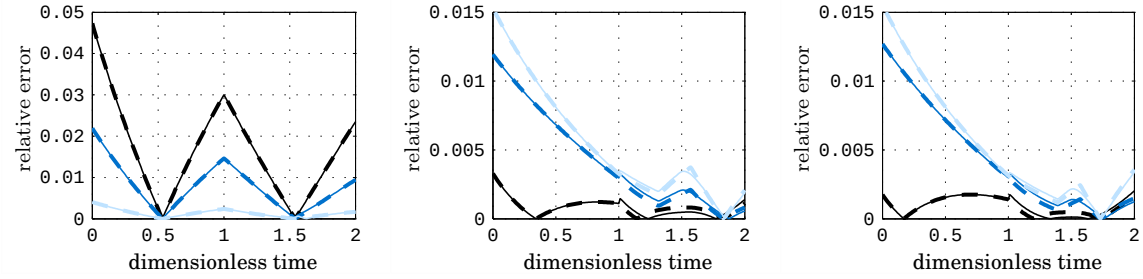

Figure 6.14.: Shape approximation of the urea simulation (page 124) with plotted absolute relative errors for the volume (black), surface area (blue) and mean width (light blue) of the projected points (bold dashed lines) and solution of the approximated problem (light lines); the left shows the objective function $f_{\mathrm{A,\,PCA}}$, the middle shows $f_{\mathrm{A,\,abs^2}}$ and the right shows $f_{\mathrm{A,\,rel\,max}}$

While the above results demonstrate that the objectives are well reflected, the results in figure 6.15 aims to demonstrate that the global optima are obtained reliably. Therefore, the objective function considers the volume ($\mu_1 = \mu^{\mathrm{volume}}$) and the mean width ($\mu_2 = \mu^{\mathrm{mean\,width}}$) while the ratio $\gamma_{\mu,1}/\gamma_{\mu,2}$ is varied between 0 and 1 in steps of 0.025. Again, errors consider the deviation between the original sample points and their projections according to equation 6.19. The initial amount of trial points in stage one of the global optimization was set to 5000 and the number of local optimization runs in stage 2 was set to 50. The corresponding computation time is less than 3 minutes for each approximation run. The left of figure 6.15 shows the resulting coordinates $b_{i,1}$ of the free variables according to equation 6.38. It indicates jumps between local optima due to the changing landscape of the objective function. The objective function for $\gamma_{\mu,1}/\gamma_{\mu,2} = 1$ is shown in the right of figure 6.13 and the indicated location of the optimum with $b_{1,1} = -0.04$ and $b_{2,1} = 0.30$ (by figure 6.15) indeed lies within the valley of the objective function. The right of figure 6.15 shows the absolute relative error of the volume (blue) and mean width (light blue) as well as the combined error according to the objective function $f_{\mathrm{A,\,rel\,max}}$ (black). The error for the volume is 3% for $\gamma_{\mu,1}/\gamma_{\mu,2} = 0$ and the corresponding local optimum changes quickly when this volume error is taken into account (at $\gamma_{\mu,1}/\gamma_{\mu,2} = 0.025$). Also for the next two computed approximations, the large volume errors affect the obtained local optimum. After these initial changes of the local optima, there is only one jump remaining at $\gamma_{\mu,1}/\gamma_{\mu,2} \approx 0.25$ where the local optimum for $\gamma_{\mu,1}/\gamma_{\mu,2} = 1$ is obtained. Despite the jumps in the free variables $b_{i,1}$, the objective function changes continuously. This would not be the case if local optima are obtained unreliably and with significantly lower parameters for the global optimization algorithm, the graphs in the right of figure 6.13 would contain corresponding sporadic

jumps.

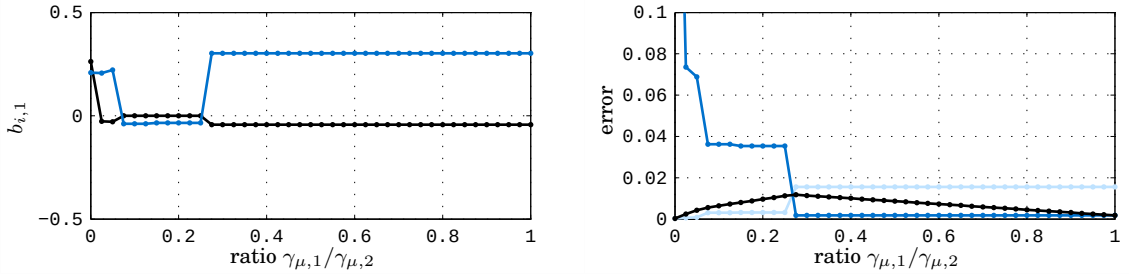

Figure 6.15.: Shape approximation of the urea simulation (page 124) based on the objective $f_{\mathrm{A,\,rel\,max}}$ with $\mu_1 = \mu^{\mathrm{volume}}$ and $\mu_2 = \mu^{\mathrm{mean\,width}}$ and varying ratio $\gamma_{\mu,1}/\gamma_{\mu,2}$; the left shows the location of the local optimum (black: $b_{1,1}$, blue: $b_{2,1}$) and the right shows the maximum relative errors for the volume (blue), the mean width (light blue) and the combined error (black)

A last detail considers the errors between the original sample points and the approximated simulation which uses the initial condition:

$$\mathbf{h}_{\mathrm{A},i}(t=0) = \mathbf{M}_{\mathbf{h}_{\mathrm{C}}\mapsto\mathbf{h}_{\mathrm{A}}}\mathbf{h}_{\mathrm{C},i}(t=0) \tag{6.44}$$

and the growth rate:

$$\mathbf{g}_{\mathrm{A}} = \mathbf{M}_{\mathbf{h}_{\mathrm{C}}\mapsto\mathbf{h}_{\mathrm{A}}}\mathbf{g}_{\mathrm{C}}. \tag{6.45}$$

These results are plotted by the thin lines in figure 6.14 and deviations from the direct projection of the sample points, discussed above, are only visible starting at $t \approx 1.5$. At this point in time, the facets that are represented by $h_{\mathrm{C},3}$ grow out (see figure 6.11). Both errors remain well comparable and the chosen approach for shape approximation is not only valid for the direct projection, but also for the approximation of the original dynamical problem (initial condition and growth rate).

6.2.3 Case Study (Paracetamol)

Motivation. The derivation of the shape approximation procedure in the previous subsection considers a relatively simple example. First, only a single particle is analyzed. Simulating particle populations creates much more sample points and the computational effort must be reconsidered. Secondly, the growth rate was chosen generically and no interaction between the growth rate and the accumulated particle volume was considered. At last, the supersaturation often depletes at the end of a batch crystallization so that the final time steps result in very similar shapes while large shape changes might be observed at earlier time intervals. In such cases, the objective functions $f_{\mathrm{A,PCA}}$ or $f_{\mathrm{A,abs^2}}$ would yield approximations in favor of the large particles at the end of the simulation, possibly neglecting important particle shapes obtained in short time intervals, only. The following simulation setup is created to study the above concerns.

Simulated process. Simulated is the growth of paracetamol in a seeded batch reactor [Borchert et al., 2007, Borchert, 2012]. The crystal system is shown in figure 6.16 and important material data is given in table 6.2, including the growth rates. Supersaturation is generated by a temperature profile $T(t)$ that starts at $40°\mathrm{C}$ and cools down to

$30°\mathrm{C}$ within $4000\,\mathrm{s}$ or roughly $1.1\,\mathrm{h}$. Afterwards, the temperature stays constant while an overall time span of $6\,\mathrm{h}$ is simulated. The original crystal system uses $n_\mathrm{C} = 4$ and the target dimension for the shape approximation is $n_\mathrm{A} = 2$.

Initial condition. The reactor is initially filled with water ($m_\mathrm{water} = 1\,\mathrm{kg}$) that is saturated with dissolved paracetamol. A total mass of seed crystals ($m_\mathrm{crystal} = 10^{-3}\,\mathrm{kg}$) is added at $t = 0$. The particle number distribution is a normal distribution with mean facet distances:

$$h_{\mathrm{C},i} = 50\,\mu\mathrm{m} \tag{6.46}$$

and a corresponding standard deviation of 0.05 times the mean width. For the simulation, the reduced numerical setup from section 4.2 is applied. The sample vectors $\mathbf{h}_{\mathrm{C},i}$ are generated randomly proportional to the given normal distribution[2] while, with the chosen mean width and standard deviation, all 1000 generated vectors are valid. The assigned particle numbers N_i for each sample point are equal to each other, given the selected sampling proportional to the number distribution. These particle numbers N_i are computed to guarantee the initial crystal mass:

$$\begin{aligned} m_\mathrm{crystal}(t=0) &= \varrho_\mathrm{paracetamol}\, I_{\mu^\mathrm{volume}}(t=0) \\ &= \varrho_\mathrm{paracetamol} \sum_i N_i \mu^\mathrm{volume}\left(\mathbf{h}_{\mathrm{C},i}\right) \end{aligned} \tag{6.47}$$

$$N_i = \frac{m_\mathrm{crystal}(t=0)}{\varrho_\mathrm{paracetamol} \sum_i \mu^\mathrm{volume}\left(\mathbf{h}_{\mathrm{C},i}\right)}. \tag{6.48}$$

Supersaturation. The solute concentration $x_\mathrm{paracetamol}$ is defined as a mole fraction and the temperature dependent solubility $x_\mathrm{solubility}(T)$ is listed in table 6.2. Since the initial supersaturation is $s = 0$, the total mass of paracetamol in the system is given by:

$$m_\mathrm{paracetamol} = m_\mathrm{crystal}(t=0) + \frac{x_\mathrm{solubility}(T(0))}{1 - x_\mathrm{solubility}(T(0))} \frac{M_\mathrm{paracetamol}}{M_\mathrm{water}} m_\mathrm{water} \tag{6.49}$$

where $M_\mathrm{paracetamol}$ and M_water are the molar masses (table 6.2). The current total crystal volume $V_\mathrm{crystal}(t)$ can be computed based on the current sample points $\mathbf{h}_{\mathrm{C},i}$ (see equation 4.16) and the current mole fraction results in:

$$x_\mathrm{paracetamol}(t) = \frac{m_\mathrm{paracetamol} - \varrho_\mathrm{paracetamol} V_\mathrm{crystal}(t)}{m_\mathrm{paracetamol} - \varrho_\mathrm{paracetamol} V_\mathrm{crystal}(t) + \frac{M_\mathrm{paracetamol}}{M_\mathrm{water}} m_\mathrm{water}}. \tag{6.50}$$

The supersaturation is finally defined by:

$$s(t) = \frac{x_\mathrm{paracetamol}(t)}{x_\mathrm{solubility}(T(t))} - 1 \tag{6.51}$$

and used to compute the growth rates according to table 6.2.

[2] This sampling is performed directly while, typically, the points would be sampled by the Metropolis algorithm from the particle number distribution $\tilde{n}$. The particle number distribution is, however, only implicitely given by the total particle weight. This information does not transfer easily to the total particle number I_1, required to scale $\tilde{n}$. The described procedure circumvents to compute a particle volume integral of the unscaled distribution and the application of the Metropolis algorithm. Furthermore, it guaranties that the initial crystal mass computed from the pivots is accurate which would, otherwise, not be possible because of the involved randomness.

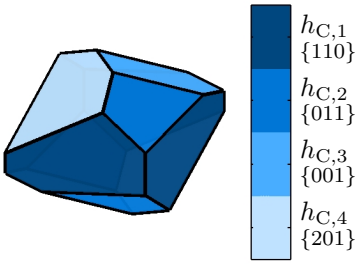

Figure 6.16.: Paracetamol crystal

Table 6.2.: Crystal data for paracetamol [Borchert et al., 2007, Borchert, 2012]

name	definition
crystal system	monoclinic, $2/m$, $a = 12.651\,\text{Å}$, $b = 8.887\,\text{Å}$, $c = 7.236\,\text{Å}$, $\beta = 114.848°$
crystal density	$\varrho_{\text{paracetamol}} = 1263\,\frac{\text{kg}}{\text{m}^3}$
molar mass	$M_{\text{paracetamol}} = 151.17\,\frac{\text{kg}}{\text{kmol}}$, $M_{\text{water}} = 18.02\,\frac{\text{kg}}{\text{kmol}}$
solubility in water	$x_{\text{solubility}}(T) = e^{\frac{10495.9\,\text{K}}{T} + 45.11344\,\ln\left(\frac{T}{\text{K}}\right) - 298.59288}\,\frac{\text{mol}_{\text{solute}}}{\text{mol}_{\text{solution}}}$
growth rate for $\{110\}$ facets	$g_{\{110\}}(s) = \frac{\text{m}}{\text{s}} \begin{cases} 1.041 \times 10^{-8} s & s \leq 0.1076 \\ 2.125 \times 10^{-6} s^2 - 2.142 \times 10^{-7} s - 4.366 \times 10^{-10} & \text{otherwise} \end{cases}$
growth rate for $\{20\bar{1}\}$ facets	$g_{\{201\}}(s) = -5.883 \times 10^{-6} s^3 + 1.847 \times 10^{-6} s^2 - 1.131 \times 10^{-9} s\,\frac{\text{m}}{\text{s}}$
growth rate for $\{011\}$ facets	$g_{\{011\}}(s) = -2.238 \times 10^{-6} s^3 + 7.461 \times 10^{-7} s^2 - 5.940 \times 10^{-8} s\,\frac{\text{m}}{\text{s}}$
growth rate for $\{001\}$ facets	$g_{\{001\}}(s) = \frac{\text{m}}{\text{s}} \begin{cases} -1.272 \times 10^{-8} + 2.431 \times 10^{-8} s & s \leq 0.0533 \\ -5.699 \times 10^{-6} s^3 + 1.867 \times 10^{-6} s^2 - 5.975 \times 10^{-8} s & \text{otherwise} \end{cases}$

Simulation results. Figure 6.17 illustrates the evolution of the mean particle shape which is computed by the mean of all generated $\mathbf{h}_\mathrm{C}$-vectors in a certain time step. A more detailed view on the shape changes is given by the evolution of the mean relative facet distances with respect to the $\{110\}$-facets ($h_{\mathrm{C},1}$) shown in the left of figure 6.18. The shape becomes more elongated in the z-direction and the $\{011\}$ facets almost grow out. The right of figure 6.18 demonstrates the increase in the total volume, surface area and mean width, relative to the corresponding initial values. The crystal mass increases by a factor of 8.5 while the particle mean width increases by the factor 2.1. The temperature profile is given in the left of figure 6.19 as an additional reference together with the supersaturation in the right.

Resulting data set. The simulation results are sampled every 60 seconds resulting in 361 snapshots for the generated $\mathbf{h}_\mathrm{C}$-vectors. Even for the PCA approach, the resulting matrix size of 4×361000 is too large to compute a shape approximation in MATLAB® [MATLAB®, 2012]. The data set must be reduced. In a first approach, the range of shapes shall be resampled evenly distributed in the $\mathbf{h}_\mathrm{C}$-space so that no specific shape is preferred. This approach could also be used in the case where no simulation data is available and only a few $\mathbf{h}_\mathrm{C}$-vectors are created manually to cover the extreme cases of shapes that might be obtained.

Even resampling. To eliminate the particle size information prior to any resampling, each $\mathbf{h}_\mathrm{C}$-vector is scaled onto a common hyperplane (see also: natively scaled structuring elements in subsection 3.3.3). The normal vector of this hyperplane is given by the mean $\mathbf{h}_\mathrm{C}$-vector from all sample vectors and the distance of that facet from the origin is 1. This step results in the scaled sample vectors:

$$\bar{\mathbf{h}}_{\mathrm{C},i} = \frac{1}{\left\langle \frac{\frac{1}{361000}\sum_i \mathbf{h}_{\mathrm{C},i}}{\left\|\frac{1}{361000}\sum_i \mathbf{h}_{\mathrm{C},i}\right\|},\ \mathbf{h}_{\mathrm{C},i} \right\rangle} \mathbf{h}_{\mathrm{C},i}. \tag{6.52}$$

Not performing this scaling would emphasize particle shapes that cover a large size range compared to particle shapes that cover only a small range of sizes. Based on the convex hull of these sample points, the covered range of $\mathbf{h}_\mathrm{C}$-vectors can be resampled by uniformly distributed $\mathbf{h}_\mathrm{C}$-vectors. The main principle of the sampling algorithm is described by Devroye [1986, section V.2.1, page 207] while the following outlines the implemented steps.

1. The sample points are expressed in an $(n_\mathrm{C}-1)$-dimensional coordinate system that is embedded in the chosen hyperplane. The required coordinate transformation is obtained by a singular value decomposition of the centered vectors $\bar{\mathbf{h}}_{\mathrm{C},i}$.

2. The convex hull and its triangulation is computed for the points in the dimensionally reduced space.

3. For each vector to be sampled a uniformly distributed scalar random variable is generated and a simplex of the triangulation is chosen proportional to the volume of that simplex. The sampled vector will reside in this simplex. The next step ensures that the vector is selected from a uniform distribution of vectors in the selected simplex. Hence, all vectors will finally be selected from a uniform distribution of vectors residing in the computed convex hull.

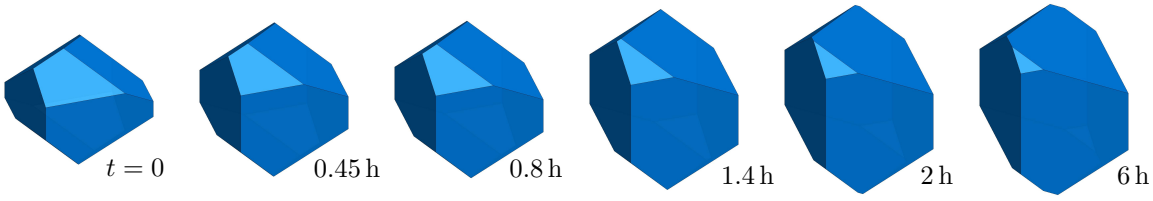

Figure 6.17.: Evolution of the mean particle shape for the paracetamol batch crystallization case study.

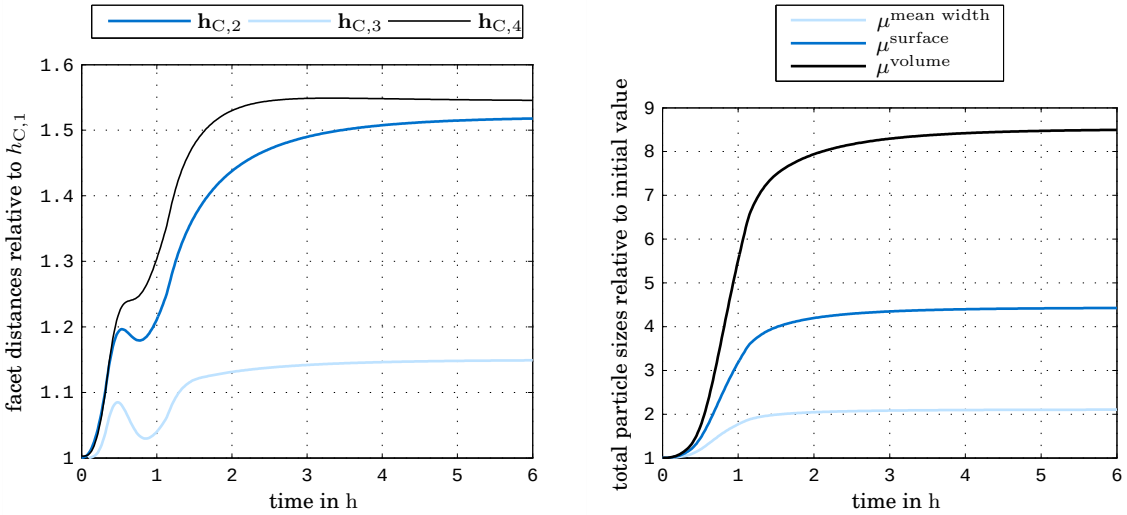

Figure 6.18.: Evolution of the relative facet distances with respect to $h_{C,1}$ (left) and increase of the total volume, surface area and mean with relative to their initial value (right) for the paracetamol batch crystallization case study.

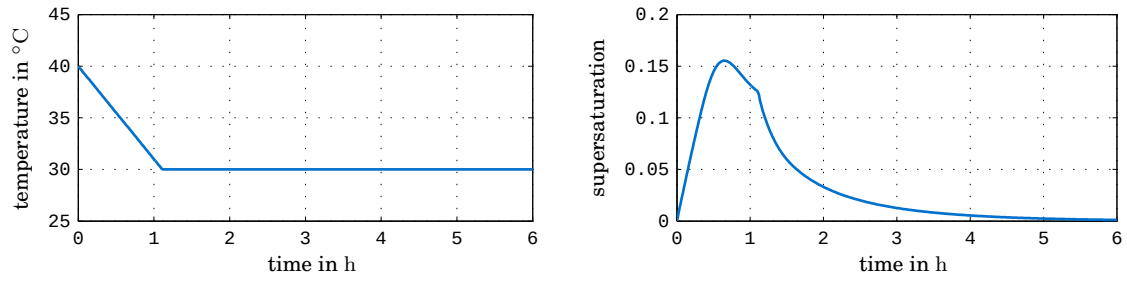

Figure 6.19.: Temperature profile (left) and supersaturation over time for the paracetamol batch crystallization case study.

4. To specify the vector in the selected simplex, it is shown in the following how uniformly distributed points are obtained for a standard simplex. Therefore, $(n_\mathrm{C}-1)$ uniformly distributed random variables $r_i \in [0,\,1]$ are generated and sorted in ascending order, prepended by $r_0 = 0$: $r_0, r_1, \ldots, r_{n_\mathrm{C}-1}$. The differences $r_i - r_{i-1}$ are also uniformly distributed so that the vector:

$$(r_1 - r_0, \ldots, r_{n_\mathrm{C}-1} - r_{n_\mathrm{C}-2})^T$$

is uniformly selected from the points of the simplex (given in $\mathcal{V}$-representation) [Devroye, 1986]:

$$\left\{ \mathbf{x} \mid x_i \geq 0, \sum_i x_i \leq 1 \right\}.$$

This standard simplex can be converted by a linear transformation into the previously selected simplex. Applying the same linear transformation to the generated vectors:

$$(r_1 - r_0, \ldots, r_{n_\mathrm{C}-1} - r_{n_\mathrm{C}-2})^T$$

results in uniformly distributed vectors for any selected simplex from step (3).

5. The linear transformation from step (1) is inverted and applied to all sample points, generated by steps (3) and (4). The range of shapes which was indicated by the sample points $\tilde{\mathbf{h}}_{\mathrm{C},i}$ are now resampled by uniformly distributed points.

Results for the even resampling. Based on the procedure above, a set of 1000 $\mathbf{h}_\mathrm{C}$-vectors was resampled from the originally 361000 $\mathbf{h}_\mathrm{C}$-vectors and a shape approximation with a reduction to $n_\mathrm{A} = 2$ was computed. Figure 6.20 illustrates relative errors for the reference PCA solution (top) and the shape approximation according to the objective $f_{\mathrm{A,rel\,max}}$ with $\mu_1 = \mu^{\mathrm{volume}}$ (bottom). The two shape approximations are compared based on two ways. The direct projection of the vectors from the original simulation: $\mathbf{h}_{\mathrm{A},i}(t) = \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{A}}\mathbf{h}_{\mathrm{C},i}(t)$ is shown by the dashed lines in figure 6.20. The solid lines show the errors for the recomputation of the simulation with the approximated shape representation. To create these simulations, the mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{A}}$ is applied to the initial condition and the growth rates and the mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{A}\mapsto\mathbf{h}_\mathrm{C}}$ is used for measure computation during the simulation. For the PCA solution, the errors for the mean facet distances do not exceed 3% for neither the projection nor the simulation error. The corresponding relative errors of the total volume, surface area and mean width are in the range of -1% and 2% for the simulation errors. The results for the optimization according to $f_{\mathrm{A,rel\,max}}$ with $\mu_1 = \mu^{\mathrm{volume}}$ took about 5 minutes while the PCA solution is obtained in a fraction of a second. Despite this high computational effort, the results for the objective $f_{\mathrm{A,rel\,max}}$ are worse. The relative errors of the facet distances even reach up to 25% so that significant changes in the mean crystal shape are visible. Figure 6.21 shows the evolution of the mean shapes based on the simulation of the approximated process model according to $f_{\mathrm{A,rel\,max}}$. Significant differences are visible compared to the mean shapes in figure 6.17.

General observation. The relative error of the total volume is typically less for the simulation error than for the projection error. This behavior is caused by the coupling of the total volume to the growth rate. If the total particle volume is overestimated by

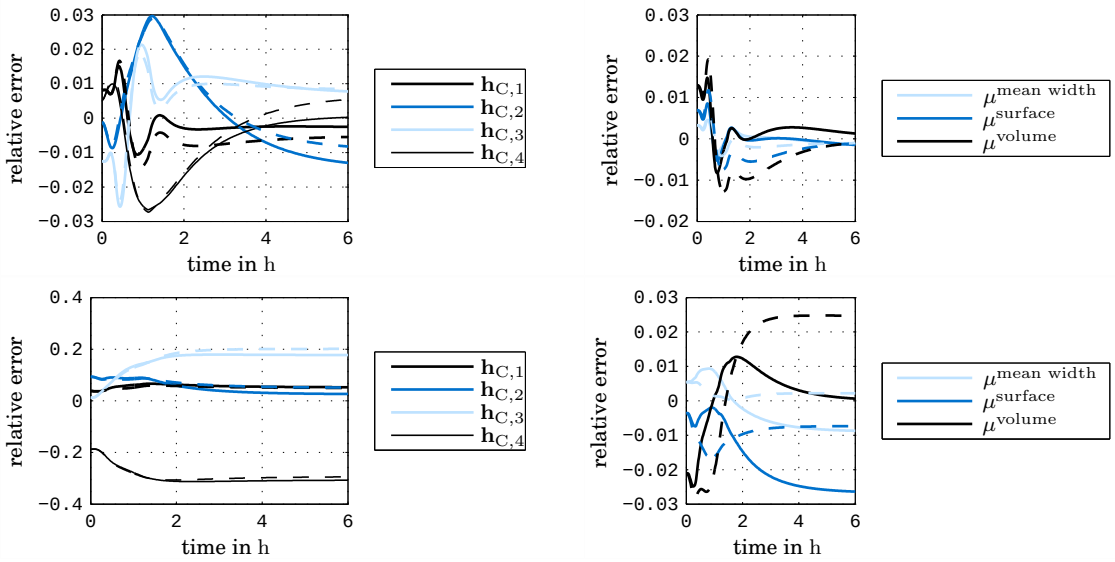

Figure 6.20.: Relative errors of the mean facet distances (left) and the total volume, surface area and mean width (right) for a shape approximation to $n_{\mathrm{A}} = 2$ based on the even resampling (top: PCA, bottom: $f_{\mathrm{A,rel\,max}}$); solid lines represent the simulation error and the dashed lines the projection error

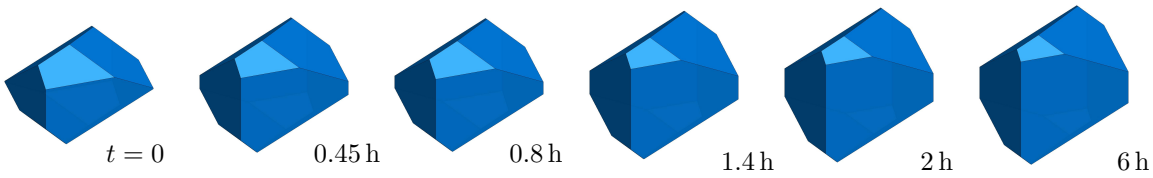

Figure 6.21.: Evolution of the mean particle shape for the shape approximation according to the objective $f_{\mathrm{A,rel\,max}}$ with $\mu_1 = \mu^{\text{volume}}$ and based on the even resampling of the simulation data

the approximation, the supersaturation and the growth rate are reduced, leading to less increase in the particle volume for further time steps. The coupling of the total particle volume and the growth rate stabilizes the obtained error for the total volume.

Discussion of the PCA solution. The reduction of the originally 4-dimensional shape approximation to 2 dimensions yields satisfying results for the PCA solution. Since characteristic sample vectors $\mathbf{h}_{\mathrm{C},i}$ which span the region for the even (re)sampling can also be obtained by experience or experiments, this approach is reasonable when no simulation data is available. However, the obtained simulation error cannot be concluded from the errors of the shape approximation. Figure 6.22 shows in the left the errors of the 1000 generated sample vectors for the PCA solution. The obtained relative errors are sorted ascendingly and span the range from -32% up to 57% for the crystal volume. Based on these large deviations it is surprising that the total particle volume does not exceed 2% error.

Discussion of the complex optimization. The projection and simulation errors for the approximation based on the objective $f_{\mathrm{A,rel\,max}}$ with $\mu_1 = \mu^{\mathrm{volume}}$ are clearly not convincing. However, this does not imply that the nonlinear optimization failed to find a suitable optimum. The right of figure 6.22 shows relative errors of the 1000 generated sample vectors that were used to find the shape approximation. The relative error of the crystal volume ranges from -35% to 30% and yields a significantly better objective value for $f_{\mathrm{A,rel\,max}}$ compared to the PCA solution in the left of the same figure. While the mean error is clearly negative, it is not possible to derive a relation between the statistics of the resampled $\mathbf{h}_{\mathrm{C}}$-vectors and the relative error of the total particle volume. To obtain a suitable shape approximation, the selection of the sample vectors and/or the selection of the objective function must be reconsidered.

Time based resampling. The even resampling was used to limit the influence of the sampled $\mathbf{h}_{\mathrm{C}}$-vectors between $t = 2\,\mathrm{h}$ and $t = 6\,\mathrm{h}$ that would otherwise dominate the shape approximation for the PCA approach. Additionally, the objective $f_{\mathrm{A,rel\,max}}$ aimed to limit the maximum relative error of the particle volume. While this limit is reasonable for a single particle (see the urea case study in subsection 6.2.2), it is not suitable for low errors of the integral $I_{\mathrm{volume}}(t)$ when a particle population is considered. In the following approach, a small amount of time steps t_i is selected from which a subset of 100 sample $\mathbf{h}_{\mathrm{C}}$-vectors is used. The time $t_1 = 0$ is selected and every time step t_{i+1} thereafter for which the mean particle shape has a mean width that is $10\,\mu\mathrm{m}$ larger than for the time step t_i before. This approach ensures that a lot of sample $\mathbf{h}_{\mathrm{C}}$-vectors are selected between $t = 0$ and $t = 2\,\mathrm{h}$ while only a few remain for $t > 2\,\mathrm{h}$. Additionally, the time $t = 6\,\mathrm{h}$ is added to the selected time steps to ensure the whole time span being covered by $\mathbf{h}_{\mathrm{C}}$-vectors. Given this sampling, the resulting complete set of $\mathbf{h}_{\mathrm{C}}$-vectors is grouped according to the time criterium t_i and the integral properties $I_{\mathrm{volume}}(t_i)$ can be evaluated based on the 100 sample points in each time step to construct the new objective:

$$f_{\mathrm{A},I_{\mathrm{volume}}} = \max_i \left| I_{\mathrm{volume}}(t_i,\, \mathbf{h}_{\mathrm{C}}(t_i)) - I_{\mathrm{volume}}(t_i,\, \mathbf{h}_{\mathrm{A}}(t_i)) \right|. \tag{6.53}$$

This new resampling of the $\mathbf{h}_{\mathrm{C}}$-vecors will be denoted by time based resampling.

Results of the time based resampling. The resampling based on the above approach generated 15 time snapshots t_i with 1500 sample $\mathbf{h}_{\mathrm{C}}$-vectors in total. Figure

6.23 illustrates relative errors for the reference PCA solution (top) and the shape approximation according to the objective $f_{\mathrm{A},I_{\mathrm{volume}}}$ (bottom), analogue to figure 6.20 that was discussed earlier. The PCA solution yields similar relative errors compared to the even resampling. However, this time the nonlinear optimization approach with the objective $f_{\mathrm{A},I_{\mathrm{volume}}}$ significantly improves the relative errors of the total volume integral below 0.5%. This relative error is about 4 times smaller than for the PCA solution. The corresponding relative errors for the components of the $\mathbf{h}_{\mathrm{C}}$-vector are much larger and reach up to 7.3% for $h_{\mathrm{C},1}$. The corresponding mean shape evolution is shown in figure 6.24 where the most apparent difference to figure 6.17 is visible in the $\{011\}$ facets corresponding to $h_{\mathrm{C},2}$.

Additional results: size distributions. It is straightfoward to compute particle size distributions from the stored variables N_i and $\mathbf{h}_{\mathrm{C},i}(t)$. Distribution values are defined by the integrals:

$$\frac{\int_{\Omega_{\mu,i}} \mu(\mathbf{h}_{\mathrm{C}})\, \tilde{n}(\mathbf{h}_{\mathrm{C}},\, t)\, \mathrm{d}\mathbf{h}_{\mathrm{C}}}{\mu_{i+1} - \mu_i} \tag{6.54}$$

where μ_i (with $\mu_i < \mu_{i+1}$) are the boundaries of disjoint bins with each sampling point at $\frac{1}{2}\left(\mu_{i+1} + \mu_i\right)$. The region $\Omega_{\mu,i}$ is defined as:

$$\Omega_{\mu,i} = \left\{\mathbf{h}_{\mathrm{C}} \mid \mu(\mathbf{h}_{\mathrm{C}}) \geq \mu_i,\, \mu(\mathbf{h}_{\mathrm{C}}) \leq \mu_{i+1}\right\}. \tag{6.55}$$

The Monte Carlo integral estimate for each distribution value is then given by:

$$\frac{1}{n_{\mathrm{sample}}} \sum_i \mu\left(\mathbf{h}_{\mathrm{C},i}(t)\right) \frac{\tilde{n}\left(\mathbf{h}_{\mathrm{C},i}(t),\, t\right)}{\tilde{w}(\mathbf{h}_{\mathrm{C},i}(t),\, t)} \delta_{\mathbf{h}_{\mathrm{C},i}(t)\in\Omega_i} \tag{6.56}$$

where $\delta_{\mathbf{h}_{\mathrm{C},i}(t)\in\Omega_i}$ is 1 for $\mathbf{h}_{\mathrm{C},i}(t) \in \Omega_i$ and 0, otherwise. The left of figure 6.25 shows particle size distributions over time for the original process model in black and the approximation to $n_{\mathrm{A}} = 2$ according to the objective $f_{\mathrm{A},I_{\mathrm{volume}}}$ in blue. The agreement is very good considering the involved large errors of the Monte Carlo estimates that can be expected for distribution plots (see equation 4.18). This example demonstrates that any computation that is available for the original process model with $n_{\mathrm{C}} = 4$ is also available for an approximation.

Additional results: optimal single shapes. Additionally, it is possible to find an optimal single shape to obtain a 1-dimensional population balance with $n_{\mathrm{A}} = 1$. Figure 6.26 illustrates corresponding relative errors for the PCA solution (top) and the shape approximation according to the objective $f_{\mathrm{A},I_{\mathrm{volume}}}$ (bottom) computed from the time based resampling. This figure is analogue to figure 6.20 or 6.23 that were discussed earlier. The resulting errors are much larger in the range of 10% for the simulation error while it is even more apparent that the coupling between the total volume and the growth rate via the supersaturation stabilizes the simulation errors compared to the projection errors. For that reason, the shape approximation according to the objective $f_{\mathrm{A},I_{\mathrm{volume}}}$ does not perform significantly better in terms of simulation errors of the total volume, even though the projection errors are better. The errors in the facet distances are very high which is also well expressed by the computed shapes in figure 6.27 when they are compared to the original mean shape evolution in figure 6.17.

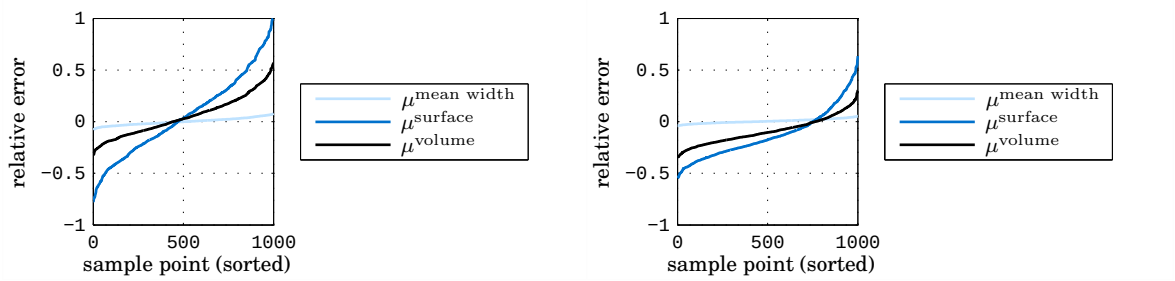

Figure 6.22.: Relative measure errors for the even resampling and f_{PCA} (left) as well as $f_{\text{A,rel max}}$ with $\mu_1 = \mu^{\text{volume}}$ (right); sample points are sorted according to the values of the relative errors

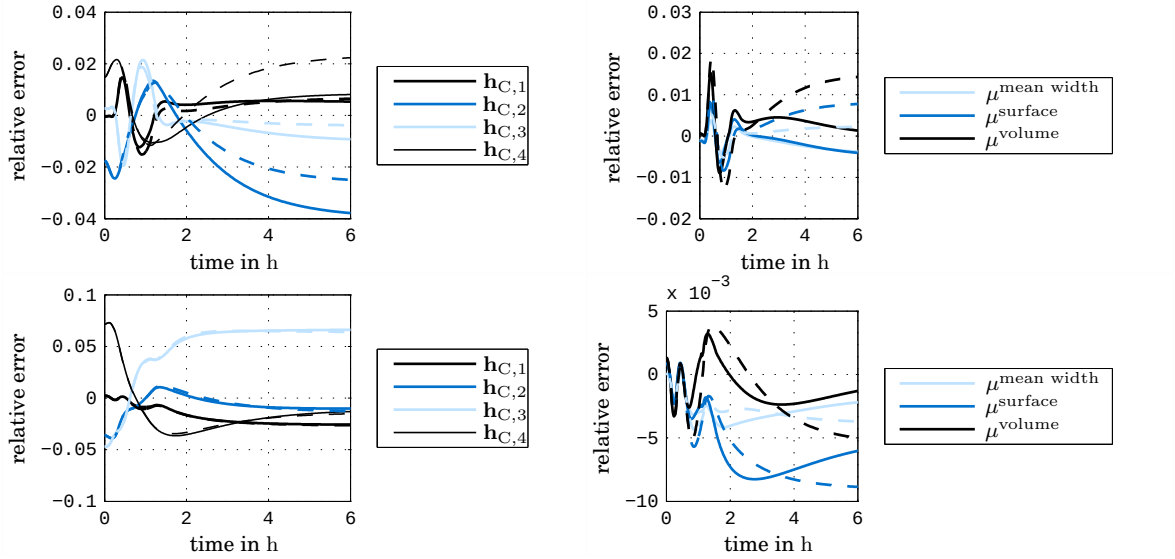

Figure 6.23.: Relative errors of the mean facet distances (left) and the total volume, surface area and mean width (right) for a shape approximation to $n_{\text{A}} = 2$ computed based on the time based resampling (top: PCA, bottom: $f_{\text{A},I_{\text{volume}}}$); solid lines represent the simulation error and the dashed lines the projection error

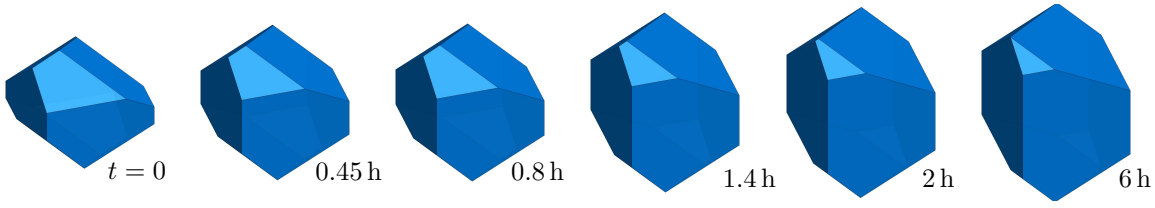

Figure 6.24.: Evolution of the mean particle shape for the shape approximation according to the objective $f_{\text{A},I_{\text{volume}}}$ and based on the time based resampling of the simulation data

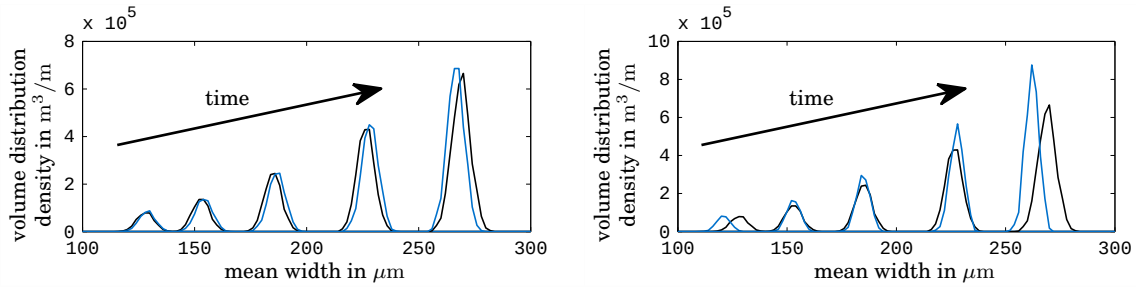

Figure 6.25.: Particle volume distributed over the total particle mean width at times $t = 0$, $0.5\,\mathrm{h}$, $0.7\,\mathrm{h}$, $1\,\mathrm{h}$ and $6\,\mathrm{h}$ for the approximation to $n_\mathrm{A} = 2$ by the objective $f_{\mathrm{A},I_\mathrm{volume}}$ (left) and to $n_\mathrm{A} = 1$ by the PCA objective (right); both shape approximations used the time based resampling.

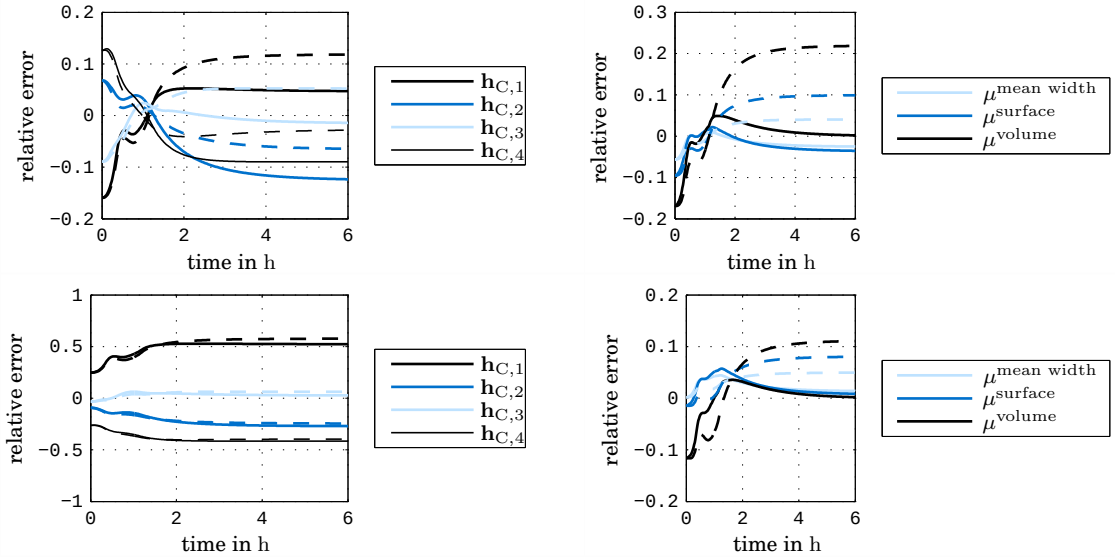

Figure 6.26.: Relative errors of the mean facet distances (left) and the total volume, surface area and mean width (right) for a shape approximation to $n_\mathrm{A} = 1$ computed based on the time based resampling (top: PCA, bottom: $f_{\mathrm{A},I_\mathrm{volume}}$); solid lines represent the simulation error and the dashed lines the projection error

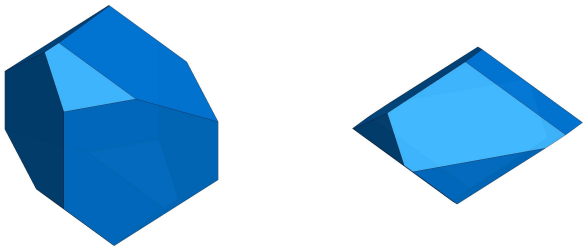

Figure 6.27.: Approximated shapes for $n_{\mathrm{A}} = 1$ computed by the time based resampling and according to the PCA objective (left) and according to the objective $f_{\mathrm{A},I_{\mathrm{volume}}}$ (right)

Applicability. Note that this approach for shape approximation will typically fail to provide reasonable results when facets disappear and reappear later in the simulated process since the validity of $\mathbf{h}_{\mathrm{C}}$-vectors is not maintained throughout a simulation. Furthermore, the resulting $\mathcal{H}_{\mathrm{C}}$-representations of a shape approximation according to this subsection are typically not proper[3] so that it will not even be possible to maintain the validity of $\mathbf{h}_{\mathrm{C}}$-vectors. While it is possible to construct a shape approximation such that the validity for specific sets of coherent facets can be maintained, this possibility will be outlined in appendix A.2.1 to A.2.3. This approach would require a detailed understanding of improper $\mathcal{H}_{\mathrm{C}}$-representations while the results that can be expected in terms of achieved reduction and accuracy of the results are questionable.

Conclusions. The shape approximation according to the PCA objective always provided a reasonable approximation of the original process model. This makes the PCA approach a reasonable reference that is easy to implement and sufficient in many cases. The shape approximations according to a tailor made objective function introduce a considerable complexity in terms of the theory, discussed in the previous subsection, and in terms of the global optimization problem that needs to be solved. The computation times are reasonable with a couple of minutes and the relative error of the total volume was reduced by a factor of 4 for the reduction to $n_{\mathrm{A}} = 2$. Additionally, the global optimization approach did not yield considerable better results for the approximation to $n_{\mathrm{A}} = 1$ since the target relative error of the total volume is stabilized automatically by its coupling to the growth rate via the supersaturation. Overall, these results imply that the global optimization is not worth the effort for similar cases. However, consider that the global optimization approach was much more efficient for the urea case study, discussed in the previous subsection. Therefore, the decision to utilize the global optimization approach remains a case by case decision and can yield significant improvements, depending on the involved range of shapes and on the focused degree of approximation (target dimension n_{A}).

[3] This is was not yet discussed but considering the $\mathbf{h}_{\mathrm{A}}$-space, it includes linear constraints ($\mathbf{M}_{\mathrm{C}}\mathbf{h} = \mathbf{0}$) in addition to the ones that are used for the original $\mathbf{h}_{\mathrm{C}}$-space. Though, this time these linear constraints do not originate from symmetry so that, typically, more groups of coherent facets are present than degrees of freedom (n_{A}); leading to improper $\mathcal{H}_{\mathrm{C}}$-representations according to condition 2 for proper representations. See subsection 3.2.2 and 3.2.3 for more details.

6.3 Reduction of Complexity

6.3.1 Reduction to Moments

Motivation. Crystallization process models typically involve the evaluation of the integral:

$$I_{\text{volume}}(t) = \int_{\mathcal{C}_{\text{V}}} \tilde{n}(\mathbf{h}_{\text{C}},\, t)\, \mu^{\text{volume}}(\mathbf{h}_{\text{C}})\, \mathrm{d}\mathbf{h}_{\text{C}} \tag{6.57}$$

for the closure of the mass balance while product properties like the total surface area or the mean shape are available based on similar integral formulations. This section focuses on a shape approximation that allows a model reduction of the n_{C}-dimensional population balance to the time derivatives of the moments:

$$M_{(i,j,k)}(t) = \int_{\mathcal{C}_{\text{V}}} \tilde{n}(\mathbf{h}_{\text{C}},\, t)\, h_{C,i} h_{C,j} h_{C,k}\, \mathrm{d}\mathbf{h}_{\text{C}} \tag{6.58}$$

$$M_{(i,j)}(t) = \int_{\mathcal{C}_{\text{V}}} \tilde{n}(\mathbf{h}_{\text{C}},\, t)\, h_{C,i} h_{C,j}\, \mathrm{d}\mathbf{h}_{\text{C}} \tag{6.59}$$

$$M_{i}(t) = \int_{\mathcal{C}_{\text{V}}} \tilde{n}(\mathbf{h}_{\text{C}},\, t)\, h_{C,i}\, \mathrm{d}\mathbf{h}_{\text{C}} \tag{6.60}$$

$$M_{0}(t) = \int_{\mathcal{C}_{\text{V}}} \tilde{n}(\mathbf{h}_{\text{C}},\, t)\, \mathrm{d}\mathbf{h}_{\text{C}}. \tag{6.61}$$

The model reduction itself does not introduce any error to the evaluation of the above indicated integrals. However, only marginal information on the size or shape distribution is available based on the above moments. Since the information on distributions is typically not used for process control, the main application for this model reduction might be model based control. The derivation of the reduction is described in the following paragraphs and the corresponding limitations are highlighted. One of these limitations considers the complexity of measure computation and can be resolved by a shape approximation which is described in subsection 6.3.2. The urea and paracetamol case studies from the previous section are continued to demonstrate the model reduction to the moments above.

Measure computation. The integral of the total volume above can be reformulated to (see equation 2.31):

$$I_{\text{volume}}(t) = \int_{\mathcal{C}_{\text{V}}} \tilde{n}(\mathbf{h}_{\text{C}},\, t) \left(\sum_{(i,j,k)} c^{\text{volume}}_{(i,j,k),p(\mathbf{h}_{\text{C}})} h_{C,i} h_{C,j} h_{C,k} \right) \mathrm{d}\mathbf{h}_{\text{C}} \tag{6.62}$$

which simplifies to:

$$I_{\text{volume}}(t) = \sum_{(i,j,k)} c^{\text{volume}}_{(i,j,k)}\, M_{(i,j,k)}(t) \tag{6.63}$$

if all $\mathbf{h}_{\text{C}}$-vectors share the same set of coefficients $c^{\text{volume}}_{(i,j,k),p(\mathbf{h}_{\text{C}})} = c^{\text{volume}}_{(i,j,k)}$ or the same unified partition $p\,(\mathbf{h}_{\text{C}})$, respectively. The total particle surface area is given by a similar

formulation, the total particle number is equivalent to M_0 and the mean shape of the particles from the size distribution is given by:

$$h_{\mathrm{C},i} = \frac{\int_{\mathcal{C}_\mathrm{V}} \tilde{n}(\mathbf{h}_\mathrm{C},\, t)\, h_{C,i}\, \mathrm{d}\mathbf{h}_\mathrm{C}}{\int_{\mathcal{C}_\mathrm{V}} \tilde{n}(\mathbf{h}_\mathrm{C},\, t)\, \mathrm{d}\mathbf{h}_\mathrm{C}} = \frac{M_i}{M_0}. \tag{6.64}$$

Dynamic evolution of moments. The time derivative of the moments is exemplified by the moment $M_{(i,j,k)}$:

$$\frac{dM_{(i,j,k)}}{dt} = \int\limits_{\mathcal{C}_\mathrm{V}} \frac{\partial \tilde{n}(\mathbf{h}_\mathrm{C},\, t)}{\partial t}\, h_{C,i} h_{C,j} h_{C,k}\, \mathrm{d}\mathbf{h}_\mathrm{C}. \tag{6.65}$$

The partial derivative of the distribution $\tilde{n}$ can be substituted by the population balance (equation 4.3):

$$\frac{dM_{(i,j,k)}}{dt} = \int\limits_{\mathcal{C}_\mathrm{V}} \left(\sum_l \sigma_l(\mathbf{h}_\mathrm{C}, t) \right) h_{C,i} h_{C,j} h_{C,k}\, \mathrm{d}\mathbf{h}_\mathrm{C} - \int\limits_{\mathcal{C}_\mathrm{V}} \left(\nabla_{\mathbf{h}_\mathrm{C}} \bullet (\tilde{\mathbf{g}}_\mathrm{C}\, \tilde{n}) \right) h_{C,i} h_{C,j} h_{C,k}\, \mathrm{d}\mathbf{h}_\mathrm{C}. \tag{6.66}$$

The left integral term is typically zero if the nucleation size is $\mathbf{h}_\mathrm{C} = \mathbf{0}$, except for M_0. If the source term σ_l expresses a seeding event, the moments can be updated accordingly by a discrete time event. For the analyzed case studies, only an initial condition is used and no source term σ_l exists so that this term is neglected in the following. The right integral can be integrated by parts:

$$\frac{dM_{(i,j,k)}}{dt} = -\int\limits_{\partial\mathcal{C}_\mathrm{V}} \left(\vec{\boldsymbol{n}} \bullet (\tilde{\mathbf{g}}_\mathrm{C}\, \tilde{n}) \right) h_{C,i} h_{C,j} h_{C,k}\, \mathrm{d}\mathbf{h}_\mathrm{C} + \int\limits_{\mathcal{C}_\mathrm{V}} (\tilde{\mathbf{g}}_\mathrm{C}\, \tilde{n}) \bullet \left(\frac{\partial \left(h_{C,i} h_{C,j} h_{C,k} \right)}{\partial \mathbf{h}_\mathrm{C}} \right) \mathrm{d}\mathbf{h}_\mathrm{C} \tag{6.67}$$

where $\partial\mathcal{C}_\mathrm{V}$ is the boundary of the validity cone and the vectors $\vec{\boldsymbol{n}}$ are corresponding outer normal vectors. The right integral term can be further reduced to the available moments if the growth rates $\tilde{\mathbf{g}}_\mathrm{C}$ are shape independent:

$$\begin{aligned} \int\limits_{\mathcal{C}_\mathrm{V}} (\tilde{\mathbf{g}}_\mathrm{C}\, \tilde{n}) \bullet \left(\frac{\partial \left(h_{C,i} h_{C,j} h_{C,k} \right)}{\partial \mathbf{h}_\mathrm{C}} \right) \mathrm{d}\mathbf{h}_\mathrm{C} &= \int\limits_{\mathcal{C}_\mathrm{V}} \tilde{n}\, \tilde{g}_{\mathrm{C},i} h_{C,j} h_{C,k} + h_{C,i} \tilde{g}_{\mathrm{C},j} h_{C,k} + h_{C,i} h_{C,j} \tilde{g}_{\mathrm{C},k}\, \mathrm{d}\mathbf{h}_\mathrm{C} \\ &= \tilde{g}_{\mathrm{C},i} M_{(j,k)} + \tilde{g}_{\mathrm{C},j}\, M_{(i,k)} + \tilde{g}_{\mathrm{C},k} M_{(i,j)} \end{aligned} \tag{6.68}$$

The left integral term (equation 6.67) cannot be reduced to a moment formulation. But if $\tilde{n} = 0$ or $\vec{\boldsymbol{n}} \bullet \tilde{\mathbf{g}} = 0$ is guaranteed for $\mathbf{h}_\mathrm{C} \in \partial\mathcal{C}_\mathrm{V}$, it is 0 and can be neglected. The limitation $\vec{\boldsymbol{n}} \bullet \tilde{\mathbf{g}} = 0$ cannot be achieved since $\tilde{\mathbf{g}}$ must remain independent on $\mathbf{h}_\mathrm{C}$ for the right integral term. The term $\tilde{n} = 0$ for $\mathbf{h}_\mathrm{C} \in \partial\mathcal{C}_\mathrm{V}$ implies that facets must not grow out. If this assumption holds, the left integral vanishes. Since this derivation can be repeated analogously for the other moments, the time derivatives result in:

$$\frac{dM_{(i,j,k)}}{dt} = \tilde{g}_{\mathrm{C},i} M_{(j,k)} + \tilde{g}_{\mathrm{C},j}\, M_{(i,k)} + \tilde{g}_{\mathrm{C},k} M_{(i,j)} \tag{6.69}$$

$$\frac{dM_{(i,j)}}{dt} = \tilde{g}_{\mathrm{C},i} M_j + \tilde{g}_{\mathrm{C},j}\, M_i \tag{6.70}$$

$$\frac{dM_i}{dt} = \tilde{g}_{\mathrm{C},i} M_0 \tag{6.71}$$

$$\frac{dM_0}{dt} = 0. \tag{6.72}$$

Summary. Equations 6.69 to 6.72 substitute the original n_C-dimensional population balance by a finite set of ordinary differential equations while equation 6.63 is used to evaluate the total particle volume or similar integrals. The limitations of this model reduction are:

1. the growth rate $\tilde{\mathrm{g}}_\mathrm{C}$ is independent on the current particle shape or size;
2. no facet disappears so that $\tilde{n}(\mathbf{h}_\mathrm{C}, t) = 0$ for $\mathbf{h}_\mathrm{C} \in \partial\mathcal{C}_\mathrm{V}$ holds;
3. all $\mathbf{h}_\mathrm{C}$-vectors reside in the same unified partition, throughout the entire simulation.

If limitation (1) is not fulfilled, the derivation is not applicable (see equation 6.68). If limitation (2) is not fulfilled, the derivation remains applicable by choosing a sufficiently large compact domain instead of the validity cone $\mathcal{C}_\mathrm{V}$ before applying the integration by parts to equation 6.67. However, the coefficients for measure computation $c_{(\cdot)}^{\mathrm{measure}}$ are not applicable outside of the validity cone so that errors in the computed integrals can be expected. Limitation (3) is typically not fulfilled by crystal models but the shape approximation in the next subsection overcomes this obstacle. A special acknowledgment goes to Carsten Choscz [2012] who conducted the first case study that involved the above model reduction for the growth of lactose crystals.

6.3.2 Extended Embedding

Motivation. The model reduction, described in the previous subsection is only applicable if all $\mathbf{h}_\mathrm{C}$-vectors during a dynamic simulation remain in the same unified partition so that the same set of coefficients $c_{(\cdot)}^{\mathrm{measure}}$ for measure computation can be used. This is typically not the case but it is possible to construct an extension of the original $\mathcal{H}_\mathrm{C}$-representation so that any $\mathbf{h}_\mathrm{C}$-vector resides in the same unified partition of a new $\mathcal{H}_\mathrm{C}$-representation. This extension is introduced in the first part of this subsection and its implications to the original process model are outlined. It turns out that the extension only shifts the original issue of different sets of coefficients $c_{(\cdot)}^{\mathrm{measure}}$ to another problem while this problem can be resolved by a suitable shape approximation.

Reviewing unified partitions. The extension is constructed based on the understanding of the geometrical state space which is recalled from chapter 3. Given a unified partition $\mathcal{C}_{\mathrm{U},p}$, the following statements are equivalent.

1. All $\mathbf{h}_\mathrm{C}$-vectors with $\mathbf{h}_\mathrm{C} \in \mathcal{C}_{\mathrm{U},p}$ use the same set of coefficients $c_{(\cdot)}^{\mathrm{measure}}$ for measure computation.
2. All structuring elements $S_i = P(\mathbf{h}_{\mathrm{C},S_i})$ with $\mathbf{h}_{\mathrm{C},S_i} \in \mathcal{C}_{\mathrm{U},p}$ (and also all polytopes $P(\mathbf{h}_\mathrm{C})$ with $\mathbf{h}_\mathrm{C} \in \mathcal{C}_{\mathrm{U},p}$) can be summed according to a Minkowski addition by just adding the corresponding $\mathbf{h}_\mathrm{C}$-vectors:

$$\sum_i \lambda_i S_i = P_\mathbf{A}\left(\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}} \sum_i \lambda_i \mathbf{h}_{\mathrm{C},S_i}\right). \tag{6.73}$$

Important to note is the polytope term $P_\mathbf{A}\left(\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}\mathbf{h}_\mathrm{C}\right)$ explicitly indicating the facet normal matrix $\mathbf{A}$ and using the $\mathcal{H}$-representation where usually the matrix $\mathbf{A}$ is omitted

and the $\mathbf{h}_\mathrm{C}$-vectors are used directly. The shape approximation in this section will use two different matrices of facet normals. Statement (2) implies that for a different unified partition $\mathcal{C}_{\mathrm{U},q}$, two structuring elements $S_i \in \mathcal{C}_{\mathrm{U},p}$ and $S_j \in \mathcal{C}_{\mathrm{U},q}$ exist for which:

$$S_i + S_j = P_\mathbf{A}\left(\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}}\left(\mathbf{h}_{\mathrm{C},S_i} + \mathbf{h}_{\mathrm{C},S_j}\right)\right) \tag{6.74}$$

does not hold. It is used in the following to construct an extended matrix of facet normals $\tilde{\mathbf{A}}$ so that equation 6.73 holds for all structuring elements (and also for all vectors $\mathbf{h}_\mathrm{C} \in \mathcal{C}_\mathrm{V}$). This implies by the equivalence to statement (1) that a single set of coefficients $c_{(\cdot)}^{\text{measure}}$ exists for the new facet distance vectors $\tilde{\mathbf{h}}$.

Extended $\mathcal{H}$-representation. Whenever two structuring elements cannot be added according to equation 6.74, the Minkowski sum $S_i + S_j$ necessarily involves a new facet that is not listed in the matrix $\mathbf{A}$ (see subsection 2.2.3 or the paragraph *'Range of the $\mathcal{S}$-representation'* on page 53). These facets are generated by the combination of two edges. To find the facet normals for all possible additional facets, it is sufficient to check all pairs of structuring elements for new facets since no new edges can be created by Minkowski addition. The resulting set of new facet normal vectors are accumulated together with the original facets in a new facet normal matrix $\tilde{\mathbf{A}}$. The matrix $\tilde{\mathbf{A}}$ describes a new $\mathcal{H}$-representation, denoted as extended $\mathcal{H}$-representation. This extended $\mathcal{H}$-representation is only temporary and the nomenclature uses a tilde to indicate corresponding variables (see column 2 of table 6.1, page 121). Computing the corresponding vector of facet distances $\tilde{\mathbf{h}}$ is straightforward by evaluating the support values:

$$\tilde{h}_i = h\left(P(\mathbf{h}_\mathrm{C}),\, \tilde{\mathbf{a}}_i\right) = \mu^{\text{support},\,\tilde{\mathbf{a}}_i}\left(\mathbf{h}_\mathrm{C}\right).$$

Based on the unified partition dependent coefficients $c_{i,p}^{\text{support},\,\tilde{\mathbf{a}}_i}$ (see subsection 3.3.5), a corresponding unified partition dependent mapping matrix can be provided:

$$\tilde{\mathbf{h}} = \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\tilde{\mathbf{h}},p}\mathbf{h}_\mathrm{C} \tag{6.75}$$

$$\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\tilde{\mathbf{h}},p} = \begin{bmatrix} c_{1,p}^{\text{support},\,\tilde{\mathbf{a}}_1} & c_{2,p}^{\text{support},\,\tilde{\mathbf{a}}_1} & \cdots & c_{n_\mathrm{C},p}^{\text{support},\,\tilde{\mathbf{a}}_1} \\ c_{1,p}^{\text{support},\,\tilde{\mathbf{a}}_2} & c_{2,p}^{\text{support},\,\tilde{\mathbf{a}}_2} & \cdots & c_{n_\mathrm{C},p}^{\text{support},\,\tilde{\mathbf{a}}_2} \\ \vdots & \vdots & \ddots & \vdots \\ c_{1,p}^{\text{support},\,\tilde{\mathbf{a}}_{\tilde{n}_\mathrm{H}}} & c_{2,p}^{\text{support},\,\tilde{\mathbf{a}}_{\tilde{n}_\mathrm{H}}} & \cdots & c_{n_\mathrm{C},p}^{\text{support},\,\tilde{\mathbf{a}}_{\tilde{n}_\mathrm{H}}} \end{bmatrix} \tag{6.76}$$

Extended embedding. The $\tilde{\mathbf{h}}$-vectors of the structuring elements are denoted by $\tilde{\mathbf{h}}_{S_i}$ and are obtained by equation 6.75. These vectors do not span the whole $\tilde{n}_\mathrm{H}$-dimensional space since the original crystal symmetry is maintained. It follows that a new $\mathcal{H}_\mathrm{C}$-representation can be identified by the singular value decomposition of the matrix $\left[\tilde{\mathbf{h}}_{S_1}, \tilde{\mathbf{h}}_{S_2}, \cdots, \tilde{\mathbf{h}}_{S_{n_\mathrm{S}}}\right]$ which computes the mapping matrix from the (new) constrained vectors to the vectors $\tilde{\mathbf{h}}$. This new constrained $\mathcal{H}_\mathrm{C}$-representation is called the extended embedding. The descriptor 'embedding' might not yet be clear and is explained after the remark on notation. All symbols for $\mathcal{H}_\mathrm{C}$-representations are adopted. Every occurrence of the subscript C is substituted by E for 'embedding'. If no such subscript exists, $/\mathrm{E}$ is appended to the subscript, reading 'in embedding space'. An overview of the adopted notation is given in the 4^{th} column of table 6.1 (page 121).

The embedding character is created by the original $\mathbf{h}_\mathrm{C}$-vectors being mapped to corresponding $\mathbf{h}_\mathrm{E}$-vectors via:

$$\mathbf{h}_\mathrm{E} = \mathbf{M}_{\tilde{\mathbf{h}}\mapsto\mathbf{h}_\mathrm{E}}\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\tilde{\mathbf{h}},p}\mathbf{h}_\mathrm{C} \tag{6.77}$$

where the matrix $\mathbf{M}_{\tilde{\mathbf{h}}\mapsto\mathbf{h}_\mathrm{E}} = \mathbf{M}^{+}_{\mathbf{h}_\mathrm{E}\mapsto\tilde{\mathbf{h}}}$ is the direct result of the singular value decomposition for the matrix $\left[\tilde{\mathbf{h}}_{S_1}, \tilde{\mathbf{h}}_{S_2}, \cdots, \tilde{\mathbf{h}}_{S_{n_\mathrm{S}}}\right]$. This implies that the original $\mathbf{h}_\mathrm{C}$-space is embedded piecewise as a linear subspaces in the $\mathbf{h}_\mathrm{E}$-space.

Utilizing the extended embedding. The dimension n_E is often much higher than n_C so that it is not feasible to compute the whole decomposition data like the validity cone or the unified partitions. But this information is not required and it is sufficient to know that all structuring elments $\mathbf{h}_{\mathrm{E},S_i}$ now fulfill equation 6.73 and are part of the same unified partition. It is feasible to assume that the positive hull of the vectors $\mathbf{h}_{\mathrm{E},S_i}$ equals a unified partition for the purpose of measure computation. The mixed volumes can be reused or recomputed for all structuring elements and converted to the coefficients $c^{\mathrm{measure/E}}_{(\cdot)}$ (see section 3.3). Additionally, note that the $\mathbf{h}_\mathrm{E}$-vectors that are computed from original $\mathbf{h}_\mathrm{C}$-vectors according to equation 6.77 reside on the boundary of the validity cone since none of the additional facets in the normal matrix $\tilde{\mathbf{A}}$ is present for the polytopes $P_\mathbf{A}(\mathbf{h}_\mathrm{C})$.

The original process model could be transferred to the extended embedding, but the corresponding growth rates are:

$$\mathbf{g}_\mathrm{E} = \frac{\mathrm{d}\mathbf{h}_\mathrm{E}}{\mathrm{d}t} = \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E},p}\,\mathbf{g}_\mathrm{C} \tag{6.78}$$

$$\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E},p} = \mathbf{M}_{\tilde{\mathbf{h}}\mapsto\mathbf{h}_\mathrm{E}}\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\tilde{\mathbf{h}},p} \tag{6.79}$$

where p must be computed based on the current state $\mathbf{h}_\mathrm{E}$ which makes the rates $\mathbf{g}_\mathrm{E}$ shape dependent. The model reduction from subsection 6.3.1 is not applicable in such a case which is resolved in the following.

Approximation. To retain a shape independent growth rate $\mathbf{g}_\mathrm{E}$, a constant mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}$ is required. The range of this matrix indicates a linear subspace of the $\mathbf{h}_\mathrm{E}$-space that is at most n_C-dimensional since $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}$ has n_C columns. The first idea might be to reuse the shape approximation concept from section 6.2 to identify an optimal linear subspace. While this approximation can find an optimal subspace, the mapping of the original process model still uses the extended embedding:

$$\mathbf{h}_\mathrm{A} = \mathbf{M}_{\mathbf{h}_\mathrm{E}\mapsto\mathbf{h}_\mathrm{A}}\mathbf{M}_{\tilde{\mathbf{h}}\mapsto\mathbf{h}_\mathrm{E}}\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\tilde{\mathbf{h}},p}\mathbf{h}_\mathrm{C} \tag{6.80}$$

and remains, hence, shape dependent. Finding an applicable mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\tilde{\mathbf{h}}}$ or $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}$ must be resolved otherwise. A simple approach is to minimize the variance of the error:

$$f_\mathrm{variance} = \sum_i \left(\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\tilde{\mathbf{h}},p(\mathbf{h}_{\mathrm{C},i})}\mathbf{h}_{\mathrm{C},i} - \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\tilde{\mathbf{h}}}\mathbf{h}_{\mathrm{C},i}\right)^T \left(\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\tilde{\mathbf{h}},p(\mathbf{h}_{\mathrm{C},i})}\mathbf{h}_{\mathrm{C},i} - \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\tilde{\mathbf{h}}}\mathbf{h}_{\mathrm{C},i}\right) \tag{6.81}$$

for some sample $\mathbf{h}_\mathrm{C}$-vectors. The solution of this optimization problem is known by the mean of the applied mapping matrices $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\tilde{\mathbf{h}},p}$:

$$\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\tilde{\mathbf{h}}} = \frac{1}{n_\mathrm{sample}}\sum_i \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\tilde{\mathbf{h}},p(\mathbf{h}_{\mathrm{C},i})}. \tag{6.82}$$

Since the sample $\mathbf{h}_\mathrm{E}$-vectors span an at most n_C-dimensional subspace, it is sufficient to identify this subspace analogue to the PCA based shape approximation to $n_\mathrm{A} = n_\mathrm{C}$. The PCA based shape approximation does not involve any shape approximation. It is already contained in the $\mathbf{h}_\mathrm{E}$-vectors that are created by the mapping:

$$\mathbf{h}_\mathrm{E} = \mathbf{M}_{\mathbf{h}_\mathrm{C} \mapsto \mathbf{h}_\mathrm{E}} \mathbf{h}_\mathrm{C} \tag{6.83}$$

$$\mathbf{M}_{\mathbf{h}_\mathrm{C} \mapsto \mathbf{h}_\mathrm{E}} = \mathbf{M}_{\tilde{\mathbf{h}} \mapsto \mathbf{h}_\mathrm{E}} \mathbf{M}_{\mathbf{h}_\mathrm{C} \mapsto \tilde{\mathbf{h}}} \tag{6.84}$$

Summary. Figure 6.28 gives an overview of the mappings and representations that are utilized throughout this subsection. On the left, the original $\mathcal{H}$- and $\mathcal{H}_\mathrm{C}$-representations are indicated by the symbols $\mathbf{A}$, $\mathbf{h}$ and $\mathbf{h}_\mathrm{C}$. The first step created an extension of the facet normal matrix, leading to an extended $\mathcal{H}$-representation ($\tilde{\mathbf{A}}$, $\tilde{\mathbf{h}}$). The mapping to this space is either exact but only piecewise linear via $\mathbf{M}_{\mathbf{h}_\mathrm{C} \mapsto \tilde{\mathbf{h}}, p}$ (equation 6.75), or it is approximated and linear via $\mathbf{M}_{\mathbf{h}_\mathrm{C} \mapsto \tilde{\mathbf{h}}}$ (equation 6.82). Based on the original structuring elements in $\tilde{\mathbf{h}}$-space, the smallest required linear subspace can be identified which is indicated by the vectors $\mathbf{h}_\mathrm{E}$. While the process model could be transformed to the $\mathbf{h}_\mathrm{E}$-space via the exact mapping, using only one set of coefficients $c_{(\cdot)}^{\mathrm{measure/E}}$, the growth rates become shape dependent. Therefore, the approximating mapping is used which allows an additional model reduction analogue to subsection 6.1.2 which creates a third $\mathcal{H}_\mathrm{C}$-representation, indicated by vectors $\mathbf{h}_\mathrm{A}$. The formulation of the process model in $\mathbf{h}_\mathrm{A}$-space finally allows the reduction to moments. Additionally, the last step could also include another shape approximation according to subsection 6.2.1, further reducing the set of required equations.

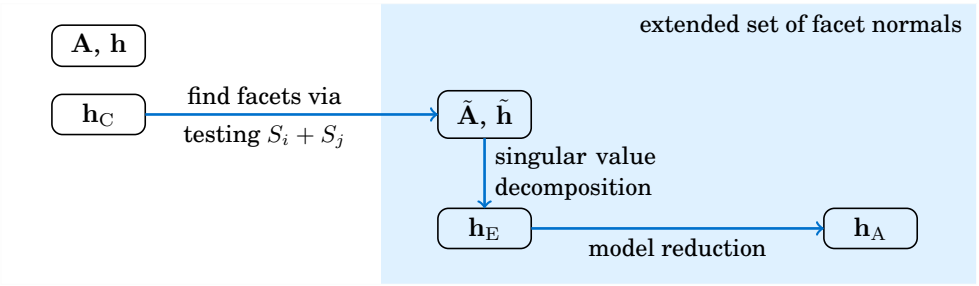

Figure 6.28.: Schematic of relevant mappings for the extended embedding

6.3.3 Case Studies

Motivation. Given the derivation of the previous subsection, it is not yet clear what an extended $\mathcal{H}$-representation looks like and how the shape approximation affects the results when the population balance is reduced to a moment model. To answer these questions, the two case studies of the previous section are continued in the following. The case study for a single growing urea crystal is introduced on page 124 and the results of the dimensional reduction are given, starting on page 131. The case study for a growing population of paracetamol crystals is introduced and analyzed in subsection 6.2.3.

Extended $\mathcal{H}$-representation. Figure 6.29 shows sample particles of the extended $\mathcal{H}$-representation in which all additional facets are visible. The facets of the original crys-

tal forms are color coded while the additional facets of the extended $\mathcal{H}$-representation are shown in white.

The original urea crystal has 3 crystal forms with 10 facets while the extended $\mathcal{H}$-representation has 4 additional facets and the extended embedding uses a 4-dimensional $\mathbf{h}_\mathrm{E}$-space. This result is feasible since urea uses 4 structuring elements with 2 unified partitions so that only two structuring elements exist that cannot be combined according to equation 6.74. The addition of these two structuring elements generates the additional 4 facets.

The original paracetamol crystal uses 4 crystal forms with 12 facets while the extended $\mathcal{H}$-representation uses 28 additional facets. The computed extended embedding uses a 9-dimensional $\mathbf{h}_\mathrm{E}$-space so that the geometrical complexity of the paracetamol crystal increases much more than for the urea crystal. This situation is reasonable since the original $\mathcal{H}_\mathrm{C}$-representation uses 11 structuring elements in 13 unified partitions and, hence, already provides a higher geometrical complexity for the original crystal representation.

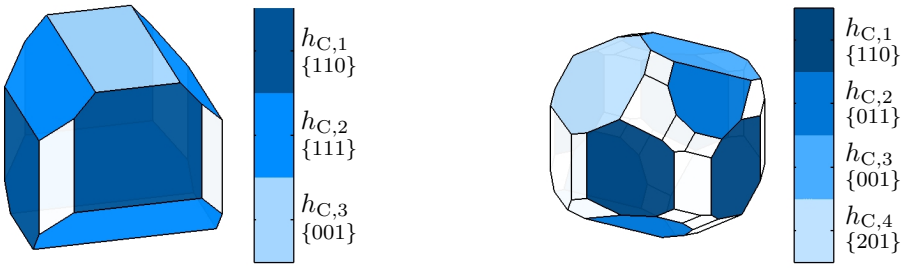

Figure 6.29.: Extended $\mathcal{H}$-representations of the urea crystal from figure 6.9 (left) and paracetamol crystal from figure 6.16 (right) with new facets in white

Generated data. The mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\tilde{\mathbf{h}}}$ was computed according to equation 6.82 which also results immediately in the mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}$ (equation 6.84). For urea, the whole series of sample points $\mathbf{h}_\mathrm{C}(t)$ is used that is generated by the non-altered process model (see page 124). For paracetamol, the $\mathbf{h}_\mathrm{C}$-vectors are taken from the time-based resampling (see page 140). The obtained $\mathbf{h}_\mathrm{E}$-vectors from the mapping $\mathbf{h}_\mathrm{E} = \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}\mathbf{h}_\mathrm{C}$ are compared to the exact mapping $\mathbf{h}_\mathrm{E} = \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E},p(\mathbf{h}_\mathrm{C})}\mathbf{h}_\mathrm{C}$ which provides a first comparison. It addresses the pure approximation error.

To obtain the n_C-dimensional $\mathcal{H}_\mathrm{C}$-representation that covers the range of the mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}$, the $\mathbf{h}_\mathrm{E}$-vectors that were used to find $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}$ are used for a PCA based shape approximation according to section 6.2. This step is, however, only a model reduction, not an additional shape approximation. The overall mapping analogue to equation 6.80 transfers any original $\mathbf{h}_\mathrm{C}$-vector to a corresponding $\mathbf{h}_\mathrm{A}$-vector while both vectors are n_C-dimensional. This mapping is applied to the initial condition of the original process model and to the growth rates which transfers the process model to the $\mathbf{h}_\mathrm{A}$-space. The initial values for the moments are then computed from the sampled points of the original simulation data.[4] The results obtained from these simulations address the final error of the obtained moment model.

[4]The urea case provides only a single sample point and the integral can be omitted.

Results for urea. At first, the more simple case of the growing urea crystal is analyzed. The relative errors for the total volume, surface area and mean width are plotted in figure 6.30. The maximum relative errors of the facet distances rated over the whole simulation time are given in table 6.3 while the error is evaluated separately for the original facets and the facets that are added by the extended $\mathcal{H}$-representation. The direct mapping into the $\mathbf{h}_\mathrm{E}$-space does not yield any error since the crystal remains in the same unified partition throughout the whole simulation. Hence, the mapping into the $\mathbf{h}_\mathrm{E}$-space does not involve a shape approximation and the model reduction could have been applied for the original problem in $\mathbf{h}_\mathrm{C}$-space. However, the simulation of the moment model yields errors in the range of up to 2%. This is solely caused by the growing out facet at $t \approx 1.55$ which can be seen by the facet distance error in table 6.3. The sudden jump in the error at $t = 1$ is caused by the accuracy of the time integrator and the switch of the growth rate at this time instant. The obtained simulation error is slightly lower compared to the shape approximation based on a dimensional reduction to $n_\mathrm{A} = 2$ (see figure 6.14).

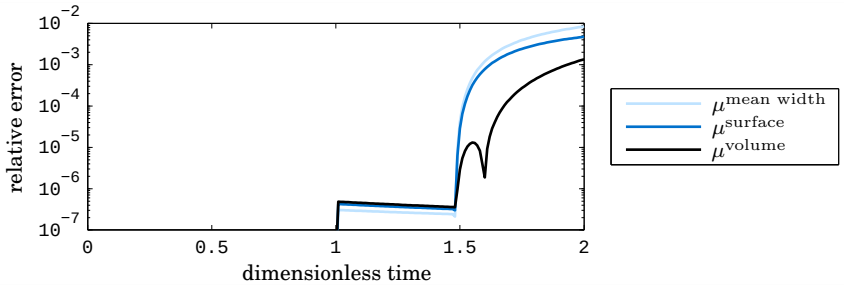

Figure 6.30.: Urea case study: relative errors of the shape approximation via the mapping into the $\mathbf{h}_\mathrm{E}$-space (dashed lines) and simulation error of the reduced moment model (solid lines)

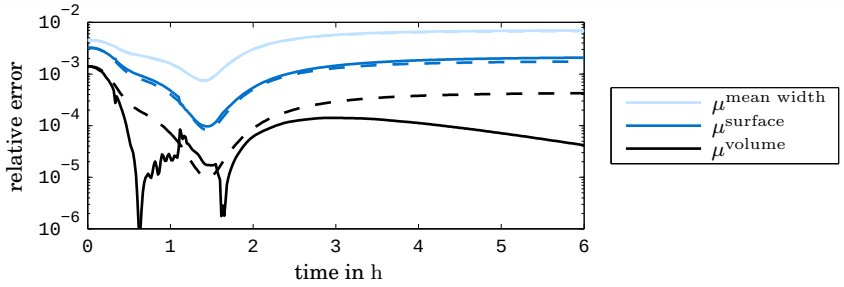

Figure 6.31.: Paracetamol case study: relative errors of the shape approximation via the mapping into the $\mathbf{h}_\mathrm{E}$-space (dashed lines) and simulation error of the reduced moment model (solid lines)

Results for paracetamol. The presentation of the results for the paracetamol case study is analogue to the urea case study above. It is given in figure 6.31 and table 6.3. The maximum relative error for the simulated moment model is obtained for the total mean width with about 0.7%. This error originates from the additional facets of

Table 6.3.: Maximum relative errors (absolute values) for facet distances of the shape approximation via the mapping into the $\mathbf{h}_\mathrm{E}$-space (mapping) and the simulation error of the reduced moment model (simulation)

	original facets	additional facets
urea (mapping)	0	0
urea (simulation)	2.3×10^{-2}	1.5×10^{-15}
paracetamol (mapping)	1.2×10^{-6}	4.2×10^{-2}
paracetamol (simulation)	4.4×10^{-4}	4.3×10^{-2}

the extended embedding that truncate the otherwise accurate crystal (see table 6.3). The original facets have a maximum relative error of 4.4×10^{-4} throughout the simulation while the additional facets have an error of 4.3%. Since these additional facets are barely present on the crystal surface, they have a minor effect on the total crystal surface area or total crystal volume. According to the small errors, changes in the mean crystal shape are not visible, if the crystals are printed at the usual size. Compared to the dimensional reduction via shape approximation (see section 6.2), the obtained relative errors for the total particle volume are significantly lower while the error of the total mean width is well comparable (see figure 6.23). However, the computational effort to obtain the shape approximation and to simulate the final process model is much lower. No more than a minute is spent for the shape approximation and the process model while no uncertainty exists whether the global optimum is found. Additionally, the original process model involves 4000 ordinary differential equations for the sample points while the moment model uses 35 ordinary differential equations.

Summary. Overall, the shape approximation into the $\mathbf{h}_\mathrm{E}$-space is well conditioned if the total particle volume is considered. The largest relative errors of the facet distances are obtained for the additionally created facets. Since these facets are not present for the original crystals, the corresponding truncation has a small effect on the crystal volume. If the moment reduction can already be performed for the original problem in $\mathbf{h}_\mathrm{C}$-space, no approximation error is involved. This is demonstrated by the urea case up to $t \approx 1.55$ or the case study on the growth of lactose crystals by Carsten Choscz [2012]. If the final set of 35 ordinary differential equations for the paracetamol case study is still too large for model based control, the step from the $\mathbf{h}_\mathrm{E}$-space to the $\mathbf{h}_\mathrm{A}$-space could involve an additional shape approximation to, *e.g.* $n_\mathrm{A} = 3$ or $n_\mathrm{A} = 2$ which would reduce the set of moments to 20 or 10, respectively.

7

Summary, Conclusions and Outlook

7.1 Summary and Conclusion

Complexity of Particle Shape Modeling. Modeling particle shape often includes two fundamental problems that are discussed in the introduction to chapter 3. The first issue concerns efficient algorithms for particle measure computation and the second issue is connected to boundaries in the geometric state space. These boundaries are the zero size boundaries at which particles disappear and validity boundaries beyond which combinations of shape coordinates are inconsistent. In the scope of crystal shape modeling, these issues are analyzed throughout chapter 3 (and appendix A.2) for constrained $\mathcal{H}$-representations. It turned out that proper $\mathcal{H}_{\mathrm{C}}$-representations for which validity boundaries can be handled consistently in simulations must be distinguished from improper representations for which inconsistent combinations of shape coordinates can appear. Several examples are used in chapter 6 to demonstrate that improper representations are valuable for model reduction techniques that often require some shape approximation[1].

Measure computation. Measure computation is discussed in detail in section 2.3 where it is shown that a strong link exists to the a-type or, in particular, to the presence of edges for 3-dimensional polytopes. Given this insight, an algorithm for Minkowski decomposition is adopted to search the geometric state space for existent a-types. The resulting domains provide linear inequalities that are used to select the right set of coefficients to compute measures based on polynomials in the coordinates of the chosen geometric state space. The volume computation for reasonably complex crystals (*e.g.* 4 forms) is then about two orders of magnitude faster compared to the computation of the volume from their $\mathcal{H}$-representation. Additionally, a large number of generic crystal systems was analyzed for the combinatorial and computational complexity that is connected to this framework of measure computation. It is found that the combinatorial complexity (*e.g.* the number of structuring elements or involved a-types) increases expo-

[1] The detailed analysis of improper representations is available in appendix A.2 where it is shown that some issues concerning the validity of geometric state vectors cannot be resolved for improper representations. This finding can, on the one hand, often be ignored but might, on the other hand, indicate severe limitations for the choice of particle shape models in general.

nentially while crystal systems with up to 9 forms are successfully tested. This implies that crystal systems with about 10 forms or more must be considered high dimensional with respect to this framework for measure computation.

Validity. The validity of state space vectors is concerned with respect to the appearance and disappearance of crystal faces. While the derived algorithm to identify valid vectors is almost identical to the algorithm by Borchert [2012], it is newly linked to present mathematical literature (see subsection 2.4.3). Additionally, a new way is proposed to maintain valid vectors during a dynamic evolution of the crystal shape. This approach also allows finding a valid vector for any given invalid vector which was not possible before. With the analysis of this validity mapping, requirements on the crystal shape representation emerged that must be fulfilled. This defines proper representations while corresponding improper representations are discussed in appendix A.2. The analysis of these improper representations is based on corresponding proper representations where validity mapping constitutes the most important tool.

Population balances. The dynamic evolution of crystal shape based on population balances is discussed in chapter 4. Since existing methods could not be implemented successfully for n-dimensional problems ($n > 2$), a numerical scheme is developed from a new approach that considers primarily the complexity of the required computation of the total particle volume instead of the complexity from the partial differential equation. It is based on Monte Carlo integral estimates that are combined with the method of characteristics. Therefore, its theoretical background is very different from existing approaches to solve population balances. The presented case studies verify this approach via a comparison to one analytical and one semi-analytical solution. Additionally, the case studies demonstrate how the stochastic degrees of freedom can be used to improve the numeric accuracy by choosing the probability distribution for the initial sample points.

Rounded particles. A new particle model for rounded particles is introduced in chapter 5 based on the newly introduced $\mathcal{S}$-representation. It describes abraded particles very well which was verified by experiments in cooperation with the group of Prof. Mazzotti (ETH Zürich, Stefan Schorsch in particular). A corresponding image analysis procedure is presented and tested for a proof of principle. The new shape model is further analyzed and required kinetic parameters are identified to describe the particle shape evolution for attrition and/or growth processes, analogue to Briesen [2009a].

Model reduction and shape approximation. Two schemes for model reduction and particle shape approximation are presented in chapter 6. The first approach aims for a dimensional reduction of the required geometric state space. It is rendered possible by the new procedure for validity mapping. In the simplest case, the corresponding optimization problem is solved by principal component analysis which often provides reasonable results. A more complex approach minimizes the error for measure computation and provides better results for specific cases, only. The second approximation approach aims for a reduction in complexity which allows the reduction of the population balance to a finite set of ordinary differential equations for moments of the size distribution. The derivation of this approach mainly utilizes the correlations between measure computation, the involved a-types and Minkowski decomposition. Both approaches to model reduction are successfully implemented and tested. They constitute

new ways to handle the complexity that arises with the shape evolution of growing crystal populations.

Conclusion of the thesis. The incorporation of mathematical literature to the analysis of the geometric state space for faceted crystals significantly improved the understanding and handling of corresponding problems. It can be concluded that measure computation and validity of the shape parameters are the two main issues for the studied system of faceted crystals. The introduction of Minkowski addition and the definition of $\mathcal{S}$-representations extend the range of shapes that can be modeled by rounded particles. Finally, the detailed analysis of the geometric state space resulted in two concepts for particle shape approximation and an understanding of the limitations that occur for dynamical particle shape modeling.

7.2 Outlook

Applications. Since this work entirely focuses on geometric aspects of crystal shape modeling, almost every analysis ends with case studies where applications with a coupling to crystallization experiments could have followed. The most apparent continuation is the compilation of crystal growth and/or attrition simulations in the scope of predictive models, process control or parameter estimation. Particularly promising is the new model for rounded particle shapes. The dynamic model and a measurement technique is ready and available to study the kinetic parameters that were isolated. Additionally, the new methods for model reduction are well suitable to establish state observers (*e.g.* Kalman filters) or approaches for model-based control.

While this work provides a solid starting point for the above applications, improvements or extensions are possible for almost every chapter of this work. The most important enhancements are listed in the remainder of this outlook.

Numerics of the Implementation. The main pitfall of the presented framework is given by the complexity of the implementation and its numerical stability. For the present work, scaling of the geometric objects was used to cover most of the issues but several workarounds are included that are necessary for errors in utilized libraries (*e.g.* the convex hull algorithm [Barber et al., 1996, MATLAB®, 2012] and the double-description method [Fukuda, 1996]). In most cases, the errors originate from the sensitivity of the algorithms to small perturbations of the input data and corresponding tolerances. To bypass these issues, the algorithms can be transferred from floating point arithmetic (*e.g.* 0.3333) to rational numbers (*e.g.* fractions like $\frac{1}{3}$). Rational numbers can be computed without numerical errors so that no tolerances are required. This is a typical approach in computational geometry and the double description method is originally implemented for rational numbers [Fukuda, 1996]. However, this approach is a trade-off between numerical stability (and accuracy) and computational efficiency since rational arithmetic is typically slower than floating point arithmetic.

Population balances. Utilizing Monte Carlo integral estimates for solution schemes of population balances raises several questions and provides new paths for population balance solvers. For the present work, only growth of an initial particle population is considered so that the main question is whether nucleation, breakage or aggregation phenomena can be incorporated. Particularly challenging is the proper handling of the probability distribution if the selection of the pivots shall be adaptive. First, some decision scheme for the removal and creation of new sample points is required. Secondly,

adding or removing sample points affects the overall probability distribution for which the integral must remain 1. If the pivots are only sampled for $t = 0$, these issues can be ignored while, in this case, a question remains that is: how to properly evaluate sink or source terms from breakage or aggregation?

Model reduction and shape approximation. Only linear subspaces are considered for approximated shapes in this work because the general understanding of the resulting improper representations was not clear in the beginning. However, nonlinear subdomains significantly improve the capabilities for a reduction of the number of shape parameters. The problem statement would optimize the parameters of a nonlinear function $f_{\mathbf{h}_\mathrm{A} \mapsto \mathbf{h}_\mathrm{C}}$ where the vectors $\mathbf{h}_\mathrm{A}$ comprise the new shape parameters. Corresponding optimization problems do not require complex concepts like the Grassmann manifolds while the selection of the right ansatz function $f_{\mathbf{h}_\mathrm{A} \mapsto \mathbf{h}_\mathrm{C}}$ might be challenging. On the other hand, finding a consistent function $f_{\mathbf{h}_\mathrm{C} \mapsto \mathbf{h}_\mathrm{A}}$ that performs the shape approximation might result in additional computational effort.

Aggregates. Since this work focuses on convex faceted particles, particle aggregates are not considered. However, particle aggregates frequently appear in crystallization processes. While the underlying convex geometry of this work is not suitable for these concave shapes, it is believed that the concept of validity and proper representations extends to general shape representations. Therefore, this work might only provide a guideline to assess possible geometric representations of particle aggregates while modeling the shape of particle aggregates including their evolution in time clearly constitutes a future milestone for crystal shape modeling.

A

Appendix

A.1 Miscellaneous

A.1.1 Mixed Volumes representing Polytope Measures

Introduction. This subsection gives additional information on how the polytope measures from subsections 2.3.2 and 2.3.3 (projection area, Feret diameter, surface area and mean width) can be interpreted as mixed volumes in n-dimensional space. These mixed volumes are formulated for the polytope itself and one or two special polytopes that are either line segments or the unit ball. The derivations here are already described in literature by [Bonnesen and Fenchel, 1934, page 37 ff]. While the discussion in this subsection is not required for the algorithms in this work, it demonstrates the concept of mixed volumes by visually expressive interpretations.

Line segments. Figure A.1 illustrates the following discussion. The polytope P is shown in blue (left) and its volume is $\mu^{\text{volume}}(P)$. The Minkowski addition of P and a unit line segment $L_{\mathbf{u}} = [\mathbf{0},\, \mathbf{u}]$ with $\|\mathbf{u}\| = 1$ is performed and the additional volume is:

$$\mu^{\text{volume}}(P + L_{\mathbf{u}}) - \mu^{\text{volume}}(P) = \mu^{\text{volume}}(L_{\mathbf{u}}) + \widetilde{V}(P,\, L_{\mathbf{u}},\, L_{\mathbf{u}}) + \widetilde{V}(P,\, P,\, L_{\mathbf{u}}). \tag{A.1}$$

The volume of the line segment is: $\mu^{\text{volume}}(L_{\mathbf{u}}) = 0$. The volume $\mu^{\text{volume}}(P + \lambda L_{\mathbf{u}})$ grows linearly with λ so that it also holds: $\widetilde{V}(P,\, L_{\mathbf{u}},\, L_{\mathbf{u}}) = 0$. The corresponding polytope $(P + L_{\mathbf{u}}) \setminus P$ is shown in light blue (middle left) and must have the volume: $\widetilde{V}(P,\, P,\, L_{\mathbf{u}})$. According to Cavalieri's principle, the volume does not change if the points of $(P + L_{\mathbf{u}}) \setminus P$ are rearranged in direction $\mathbf{u}$ to a polytope with constant width in direction $\mathbf{u}$. A prism of height 1 is obtained that is drawn in light blue (middle right). According to the transformations, the base of the prism equals the projection of P into direction $\mathbf{u}$ and the projection area is:

$$\mu^{\text{projection area}}(P,\, \mathbf{u}) = \widetilde{V}(P,\, P,\, L_{\mathbf{u}}). \tag{A.2}$$

A second line segment $L_{\mathbf{v}}$ perpendicular to $L_{\mathbf{u}}$ can be chosen with $\|\mathbf{v}\| = 1$ and the same argumentation for the 2-dimensional subspace of the projection area can be used to obtain:

$$\mu^{\text{Feret diameter}}(P,\, \mathbf{u} \times \mathbf{v}) = \tilde{V}\,(P,\, L_{\mathbf{v}},\, L_{\mathbf{u}})\,.$$

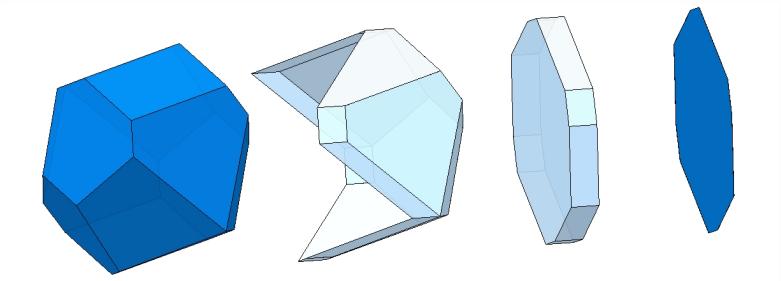

Figure A.1.: From left to right: arbitrary polytope, additional volume by Minkowski addition with line segment, transformed additional volume, projection.

The projection in direction $\mathbf{u}$ is projected onto another perpendicular plane with normal vector $\mathbf{v}$ so that the result of this projection is a line segment in direction $\mathbf{u} \times \mathbf{v}$ whose length is the Feret diameter of S in direction $\mathbf{u} \times \mathbf{v}$.

Unit ball. The mixed volumes between a convex body P and the unit ball are linked to the volume, surface area, mean width and the Euler characteristic[1] by the following relations:

$$\mu^{\text{volume}}(P) = \widetilde{V}(P, P, P) \tag{A.3}$$

$$\mu^{\text{surface area}}(P) = n\,\widetilde{V}(P, P, B) \tag{A.4}$$

$$\mu^{\text{mean width}}(P) = \frac{3}{2\pi}\,\widetilde{V}(P, B, B) \tag{A.5}$$

$$\mu^{\text{Euler characteristic}}(P) = \frac{3}{4\pi}\,\widetilde{V}(B, B, B) = 1. \tag{A.6}$$

These findings are not trivial and are connected to the mathematical terms: 'quermassintegral', 'Minkowski functional', 'intrinsic volume' and 'mixed volume' which originate from both integral and convex geometry. Details are skipped but can be found in [Schneider, 2008] with the connection between the first three terms on page 210 and the connection to mixed volumes on page 290. Alternatively, [Bonnesen and Fenchel, 1934, page 37 ff] can be used.

A.1.2 Converting between $\mathcal{H}$- and $\mathcal{V}$-representation

Motivation. According to the Minkowski-Weyl theorem there is a uniquely defined (non-redundant) $\mathcal{V}$-representation for any (non-redundant) $\mathcal{H}$-representation of a polytope and *vice versa*. The conversion from $\mathcal{H}$- to $\mathcal{V}$-representation (and *vice versa*) is frequently applied for algorithms in this work. Crystals will be given in $\mathcal{H}$-representation, while measure calculation relies on the $\mathcal{V}$-representation. The domains for $\mathbf{h}$-vectors that are introduced in chapter 3 are given in a $\mathcal{H}$-representation while the structuring elements that are used correspond to a $\mathcal{V}$-representation of the same domains.

Algorithms. The double description method [Motzkin et al., 1953] that is applied in an implementation by Fukuda [2008] provides a conversion from $\mathcal{H}$- to $\mathcal{V}$-representation. An outline of this algorithm is provided at the end of this subsection. A conversion

[1] The Euler characteristic is an integer valued measure of the topology of the body under consideration. It is an expression of the total number of components and holes. As the body under consideration is convex, the Euler characteristic is always 1.

from $\mathcal{V}$- to $\mathcal{H}$-representation is performed by the quick hull algorithm that is discussed in subsection 2.3.1. However, as the coupling to MATLAB® does not return the $\mathcal{H}$-representation and the double description method is said to be more efficient for higher dimensions [Barber et al., 1996], the double description method is applied in most of the cases.[2] Nevertheless, both algorithms could be used to perform both conversions, given the following principle of duality.

Duality. Duality is a concept from the field of combinatorial geometry [Edelsbrunner, 1987]. It provides a one-to-one correspondence between points and hyperplanes [Edelsbrunner, 1987, section 1.6]. Given a valid theorem about the combinatorics of hyperplanes and points, this theorem can be transformed to a different valid theorem by exchanging every occurrence of the word 'hyperplane' by the word 'point' and *vice versa*. A simple example is the following dual pair.

- A point is uniquely defined by n intersecting hyperplanes.
- A hyperplane is uniquely defined by n linearly independent points.

In general, there are some more details to consider (*e.g.* the term 'intersecting') when translating theorems to their dual version. The following example is essential for the application of duality in this section.

- A polytope is uniquely defined by its extreme points ($\mathcal{V}$-representation)
- A polytope is uniquely defined by its facets ($\mathcal{H}$-representation)

Duality transformation. The concept of duality does not require a specific transformation between points and hyperplanes to be defined. More than one of such transformations exists from which the following is commonly applied. It is based on the unit hypersphere. Figure A.2 (left) illustrates the shape that was already used throughout subsection 2.1.2 (see figure 2.4) together with the unit hypersphere (black circle). Right of this shape is drawn the dual polytope, again together with the unit hypersphere. The facet with normal vector $\mathbf{a}$ (left) intersects the unit circle so that the dual point $\mathbf{r}'$ lies outside of the unit circle while the direction is maintained (right, dashed line). The extreme point $\mathbf{r}$ (left) is outside of the unit circle and much farer away than the point $\mathbf{r}'$ (right). Therefore, the facet with normal vector $\mathbf{a}'$ (right) intersects the unit sphere closer to its center. The definition of this duality transformation is given by:

$$\mathbf{a}' = \mathbf{r},\ h' = 1 \tag{A.7}$$

for arbitrary points $\mathbf{r} \neq \mathbf{0}$ and:

$$\mathbf{r}' = \mathbf{a}/h \tag{A.8}$$

for an arbitrary facet normal $\mathbf{a}$ and facet distances $h \neq 0$. The facet normals $\mathbf{a}'$ are not necessarily scaled to $\|\mathbf{a}'\| = 1$. If the distance of the hyperplane from the origin is smaller (or greater) than 1, the dual point is outside (or inside) of the unit sphere like described in the example above.

[2] Unfortunately, the implementation of the double description method suffers from rare instabilities that result in a failure of the conversion. Therefore, this method is backed up by the convex hull algorithm in case of failure.

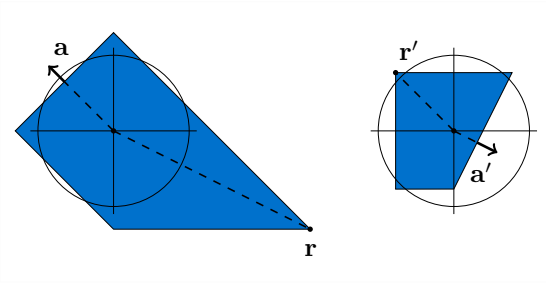

Figure A.2.: Two dual polytopes where the black circle represents the unit circle

Equation A.7 can be used to convert a polytope from $\mathcal{V}$-representation to a different polytope in $\mathcal{H}$-representation. This resulting polytope is the dual polytope under the given duality transformation. Likewise, equation A.8 can be used to obtain the dual polytope in $\mathcal{V}$-representation from a polytope in $\mathcal{H}$-representation. Note also that the dual of the dual is the original polytope, which is also evident by equations A.7 and A.8.

Application of duality. It follows that for every well defined polytope, there exists a dual polytope, while 'well defined' refers to the restrictions for equations A.7 and A.8. These are fulfilled whenever the origin is in the interior of the polytope. This is ensured by a scaling of the input polytopes that will be explained in subsection A.1.3.

Given an algorithm for the conversion from $\mathcal{H}$- to $\mathcal{V}$-representation, *e.g.* the double description method, a polytope in $\mathcal{V}$-representation can be converted to $\mathcal{H}$-representation. The required steps are illustrated by figure A.3 and given in the following.

1. Transfer the given polytope in $\mathcal{V}$-representation into the dual polytope in $\mathcal{H}$-representation by equation A.7 (Figure A.3, transition from the top right to the bottom right).

2. Perform the double description method that returns the corresponding dual $\mathcal{V}$-representation (Figure A.3, transition from the bottom right to the bottom left).

3. Use equation A.7 again to obtain the $\mathcal{H}$-representation of the original polytope (Figure A.3, transition from the bottom left to the top left).

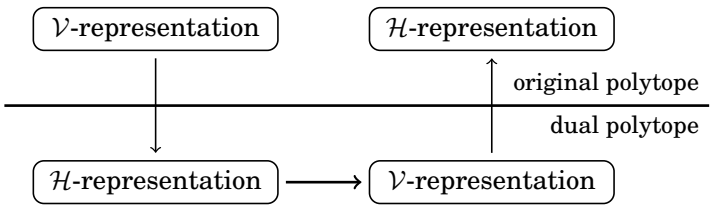

Figure A.3.: Conversion from $\mathcal{V}$- to $\mathcal{H}$-representation via duality transformations (vertical transitions) and a conversion from $\mathcal{H}$- to $\mathcal{V}$-representation (horizontal transition)

Double description method. The double description method as described by Fukuda [1996] converts an n'-dimensional convex cone:

$$C = \{\mathbf{x} \mid \mathbf{A}'\,\mathbf{x} \leq \mathbf{0}\} \tag{A.9}$$

given by the matrix $\mathbf{A}'$ to its generating rays where each ray (called generator) is represented by a point $\mathbf{r}'_i \neq \mathbf{0}$. The whole set of generating rays is represented by a matrix: $\mathbf{R}' = (\mathbf{r}'_1, \ldots, \mathbf{r}'_{n_\mathrm{E}})^T$. The same cone as in equation A.9 is then described by:

$$C = \{\mathbf{x} \mid \mathbf{x} = \mathbf{R}'\,\boldsymbol{\lambda},\, \boldsymbol{\lambda} \in \mathbb{R}^{n_\mathrm{E}},\, \lambda_i \geq 0\} \tag{A.10}$$

analogue to the positive hull: $C = \mathrm{pos}\,\{\mathbf{r}'_1, \ldots, \mathbf{r}'_{n_\mathrm{E}}\}$. That the matrix $\mathbf{R}'$ exists, given an arbitrary matrix $\mathbf{A}'$, is covered by the Minkowsky-Weyl theorem. The algorithm of the double description method itself is based on a theorem that states that for a given double description (matrices $\mathbf{A}'$ and $\mathbf{R}'$), the matrix $\mathbf{A}'$ can be extended with an additional constraint and a new double description can be obtained by updating the matrix $\mathbf{R}'$ in a manner that is defined by the theorem. This constitutes the core of an iterative algorithm. More details can be found in [Fukuda, 1996].

Double description method for polytopes. The problem that is solved by the double description method above is equivalent to finding the $\mathcal{V}$-representation for a $\mathcal{H}$-representation of a given polyhedron. This is shown in the following. It is assumed that the n-dimensional polytope is embedded in the hyperplane $\{\mathbf{x} \mid \langle \mathbf{x}, \mathbf{1}\rangle = 1\}$ of an $(n+1)$-dimensional space so that the corresponding double description is ($n' = n + 1$):

$$\mathbf{A}' = (\mathbf{h}, -\mathbf{A}) \tag{A.11}$$

$$\mathbf{R}' = \begin{pmatrix} 1 & \mathbf{x}_{\mathrm{E},1} \\ \vdots & \vdots \\ 1 & \mathbf{x}_{\mathrm{E},n_\mathrm{E}} \end{pmatrix}. \tag{A.12}$$

It is assumed that the set of generating rays is minimal so that the total number of rays equals the total number of extreme points n_E.

A.1.3 Scaling of Polytopes

Motivation. Often in the algorithms of this work, it is required to decide if a point resides on a hyperplane or if two hyperplanes or points coincide. Because of numerical inaccuracy and varying sizes of the polytopes, a relative tolerance is desirable. This is realized by scaling the polytopes and applying an absolute tolerance to the decision statements. Additionally, the applied algorithms may have some assumption that must be ensured. For example, applying the duality for an $\mathcal{H}$-representation (see equation A.8), the facet distances must not be 0. This results in the requirement of a normalized position. Hence, scaling the size and normalizing the position of polytopes is crucial to control numerical accuracy and to ensure assumptions of underlying algorithms. The following paragraphs present the scaling that is applied in this work for the $\mathcal{H}$- and for the $\mathcal{V}$-representation.

$\mathcal{H}$-representation. The largest ball that is completely contained in a given bounded set is known as the Chebychev ball. The center $\mathbf{x}_0$ and radius r of this ball are well

defined and can be found by a linear optimization problem that maximizes r by varying $\mathbf{x}_0$ and r. A ball is inscribed in the given polyhedron, if it holds for all facets i:

$$\max_{\mathbf{u} \in S^{n-1}} \left(\mathbf{a}_i^T \left(\mathbf{x}_0 + r\, \mathbf{u} \right) \right) \leq h_i. \tag{A.13}$$

Only the points of the unit sphere S^{n-1}, which is the boundary of the unit ball, are considered because of the convexity of the ball and the polytope. Since the maximum is always obtained for $\mathbf{u} = \mathbf{a}_i$, this condition can be rewritten as:

$$\mathbf{A}\, \mathbf{x}_0 + r \leq \mathbf{h} \tag{A.14}$$

which represents the set of linear constraints for the optimization problem. The optimization problem to maximize r is then solved by the simplex algorithm, readily available in MATLAB® [2012]. The original polytope is then scaled to:

$$\mathbf{h}' \;=\; \frac{1}{r} \left(\mathbf{h} - \mathbf{A}\, \mathbf{x}_0 \right) \tag{A.15}$$

where the matrix of facet normals $\mathbf{A}$ remains constant. The Chebychev ball for a scaled polytope uses $\mathbf{x}_0 = \mathbf{0}$ and $r = 1$.

$\mathcal{H}$-representation (interpretation). Equation A.15 suggests that a Chebychev radius of 0 is invalid for the scaling of polytopes. Indeed, such a radius indicates a flat polytope that could be embedded in a lower dimensional linear subspace. Flat polytopes remain, therefore, unscaled and the calling algorithm must handle flat polytopes appropriately. A Chebychev radius of ∞ indicates an unbounded polytope for which the Chebychev ball is not defined and the polytope cannot be properly scaled. Again, the calling algorithm must consider unbounded polytopes appropriately.

The scaled polytope has its Chebychev center in the origin so that the origin is an inner point of the polytope. The corresponding Chebychev radius is 1 which implies $h_i \geq 1$ for all facets. However, large facet distances are obtained for elongated or flat polytopes so that these cases are prone for numerical problems in subsequent calculations.

$\mathcal{V}$-representation. By duality, the above approach to scale a $\mathcal{H}$-representation could be translated to an approach for scaling a $\mathcal{V}$-representation. In this case, the Chebychev ball becomes the smallest ball that contains the polytope. However, a much simpler approach can be used, based on the rectangular bounding box. The bounding box is a rectangular box with edges parallel to the coordinate axes while every facet of the bounding box touches the polytope. It can be written in a $\mathcal{H}$-representation by:

$$\mathbf{A}_{\text{bounding box}} \;=\; \begin{pmatrix} \mathbf{I}_n \\ -\mathbf{I}_n \end{pmatrix} \tag{A.16}$$

$$\begin{aligned} \mathbf{h}_{\text{bounding box}} \;=\; \Big(& \max_i x_{\mathrm{E},i,1},\, \ldots,\, \max_i x_{\mathrm{E},i,n}, \\ & -\min_i x_{\mathrm{E},i,1},\, \ldots,\, -\min_i x_{\mathrm{E},i,n} \Big)^T . \end{aligned} \tag{A.17}$$

where $x_{\mathrm{E},i,j}$ is the j-th coordinate of the i-th extreme point. The scaling of the original $\mathcal{V}$-representations with the vertices in $\mathbf{R}$ is executed in two steps:

$$\mathbf{R}' = \left(\mathbf{R} - \begin{pmatrix} \frac{\max_i x_{\mathrm{E},i,1} - \min_i x_{\mathrm{E},i,1}}{2} & \cdots & \frac{\max_i x_{\mathrm{E},i,n} - \min_i x_{\mathrm{E},i,n}}{2} \\ \vdots & \ddots & \vdots \\ \frac{\max_i x_{\mathrm{E},i,1} - \min_i x_{\mathrm{E},i,1}}{2} & \cdots & \frac{\max_i x_{\mathrm{E},i,n} - \min_i x_{\mathrm{E},i,n}}{2} \end{pmatrix} \right) \tag{A.18}$$

$$\mathbf{R}'' = \mathbf{R}' \begin{pmatrix} \frac{1}{\max_i x_{\mathrm{E},i,1} - \min_i r_{i,1}} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & 0 & \frac{1}{\max_i x_{\mathrm{E},i,n} - \min_i x_{\mathrm{E},i,n}} \end{pmatrix} \tag{A.19}$$

where the first step centers the polytope at the origin of the coordinate system and the second step normalizes the size of the polytope. The polytope with the extreme points $\mathbf{R}''$ has a maximum for the absolute coordinate values that is 1.

$\mathcal{V}$-representation (interpretation). In comparison to the scaling of a $\mathcal{H}$-representation, the scaling of a $\mathcal{V}$-representation is less problematic. First, a $\mathcal{V}$-representation cannot be unbounded. Secondly, even flat polytopes can be scaled as long as they are not well aligned to one of the coordinate axes ($\max_i x_{\mathrm{E},i,j} - \min_i x_{\mathrm{E},i,j} = 0$). However, flat polytopes will contain extreme points that are relatively close so that these polytopes are prone to numerical instabilities when, for example, edge lengths of a 3-dimensional polytope are considered.

A.1.4 Measuring Roundness based on Multiple Projections

Motivation. In subsection 5.3.3, the two available orthogonal projections of the particles are analyzed separately. Averaging the results of both images, however, improves the accuracy by reducing the effect of random errors. Some theoretical background for this averaging is considered in the following while a more general case is assumed that uses an arbitrary number of projection directions that are not necessarily orthogonal to each other.

Scope. When n_{proj} projections Π_k are available, the parameters $\mathbf{h}_{\mathrm{k},\Pi_k}$ and $\lambda_{\mathrm{r},\Pi_k}$ of the projected shape model (see equation 5.39) can be obtained by the procedure, described in subsection 5.3.1. This results in a mean width $\mu_{\Pi_k}^{\text{mean width}}$ and a roundness measure $\mu_{\Pi_k}^{B1}$ for each projection. The aim is, however, to estimate the mean width and roundness μ^{B1} for the 3-dimensional particle for which the estimators are given in the following.

Mean width. The mean width $\mu_{\Pi_k}^{\text{mean width}}$ from each 2-dimensional projection (computed analogously to equation 5.6) is also an estimate for the mean width of the 3-dimensional particle. In general, the mean width of the particle can be estimated by:

$$\mu^{\text{mean width}} \approx \sum_k w_k^{\text{mean width}} \mu_{\Pi_k}^{\text{mean width}} \tag{A.20}$$

with appropriate positive weights $w_k^{\text{mean width}}$ that fulfill $\sum w_k^{\text{mean width}} = 1$. The weighting $w_k^{\text{mean width}} = \frac{1}{n_{\mathrm{proj}}}$ is usually reasonable. It is also the proper choice for the used measurement setup in subsection 5.3.3. However, for arbitrary projection directions the values of $\mu_{\Pi_k}^{\text{mean width}}$ can be stochastically dependent. For example, two projection directions that are close to each other result in similar estimates for the mean width.

If this is the case, proper weights $w_k^{\text{mean width}}$ can be computed by a Voronoi tessellation on the surface of a unit sphere (see: [Serra, 1982, Ohser and Mücklich, 2000, Schnaitter, 2011]). Additionally, the mean width estimate from a single projection might be biased when particles are not properly randomly oriented with respect to the projection direction. For example, elongated particles might take a preferred orientation so that a 2-dimensional estimate of the mean width becomes biased. For such cases, the aforementioned proper weights also guarantee an unbiased particle mean width when a sufficient set of projection directions is used.

Roundness. Given the relation between the 3-dimensional shape model and its projection, each estimated radius $\lambda_{\mathrm{r},\Pi_k}$ is also an estimate for the radius λ_{r}. Therefore, the mean width based additive roundness of the particle can be estimated by:

$$\mu^{B1} \approx \frac{2\sum_k w_k^{\lambda_{\mathrm{r}}} \lambda_{\mathrm{r},\Pi_k}}{\mu^{\text{mean width}}} \tag{A.21}$$

where the weights $w_k^{\lambda_{\mathrm{r}}} = \frac{1}{n_{\mathrm{proj}}}$ are a reasonable choice. Only when the radii $\lambda_{\mathrm{r},\Pi_k}$ have a different accuracy or their results are stochastically dependent, the weights $w_k^{\lambda_{\mathrm{r}}}$ should be reconsidered.

A.2 Improper $\mathcal{H}_{\mathrm{C}}$-representations

A.2.1 Proper Embedding

Motivation. The $\mathcal{H}_{\mathrm{C}}$-representations corresponding to approximated $\mathbf{h}_{\mathrm{A}}$-vectors are typically not proper, even though these improper representations are already successfully utilized in the preceding sections of this chapter. The two typical issues for particle shape modeling (see subsection 3.1.1) are properly resolved or neglected. The first issue, measure computation, is resolved by restoring $\mathbf{h}_{\mathrm{C}}$-vectors of the original crystal representation where measures can be computed as usual. The second issue, the validity of $\mathbf{h}_{\mathrm{C}}$-vectors, is either neglected completely while corresponding errors are accepted (see the moment models from subsection 6.3.2) or validity is restored based on the linked $\mathbf{h}_{\mathrm{C}}$-vectors (see the dimensional reduction of section 6.2). This also implies that shape approximation will fail to produce reasonable results when facets disappear and reappear later in a simulated process. To resolve this limitation, improper $\mathcal{H}_{\mathrm{C}}$-representations must be considered in detail.

An alternative motivation is given, assuming that more arbitrary shape representations shall be constructed (not necessarily for crystals). Based on the current state of the art, one could either use proper $\mathcal{H}_{\mathrm{C}}$-representations, neglect shape changes completely or construct custom shape models for particular problems that utilize a simple geometry like cuboids or ellipsoids. Here, the analysis of improper $\mathcal{H}_{\mathrm{C}}$-representations paves the way for more general shape representations and issues for improper $\mathcal{H}_{\mathrm{C}}$-representations might be analogously present for other shape representations.

Remark on generality. The analysis in this appendix is a continuation of chapter 3 where no initial assumptions on $\mathcal{H}_{\mathrm{C}}$-representations are used. This implies foremost that the improper representations in this appendix do not necessarily originate from a shape approximation. Additionally, it is important to recall that improper representations cannot represent the symmetry of crystals so that examples might appear odd in the context of modeling crystal shapes.

Example. To highlight the most important issues for improper representations, the following example is provided. It uses the facet normal matrix (identical to equation 3.8):

$$\mathbf{A} = \begin{bmatrix} 1 & -1 & 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ 0 & 0 & 1 & -1 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}^T \tag{A.22}$$

which is also used for the examples on pages 38 and 41. The group mapping matrix in this example is:

$$\mathbf{M}_{\mathbf{h}_{\mathrm{C}} \mapsto \mathbf{h}} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 2 & 1 & 1 \end{bmatrix}^T \tag{A.23}$$

and figure A.4 illustrates the most important transitions. From the left to the right of this figure, $h_{\mathrm{C},1}$ remains constant while $h_{\mathrm{C},2}$ increases. In the left sample shape, the square facets have just grown out. This part illustrates that the square facets could be corrected when they are invalid. Towards the right of figure A.4, the top right and bottom left facets of the cross polytope disappear before the other two facets of the cross polytope. This indicates that the top right and bottom left facets cannot be fixed when they become invalid. However, the top left and bottom right facets could be fixed. Providing a scheme to distinguish situations where invalid facets can be fixed from situations for which this is not possible is not straightforward. Additionally determining a required mapping procedure in $\mathbf{h}_{\mathrm{C}}$-space is similarly difficult since validity mapping (see subsection 2.4.3) is not applicable.

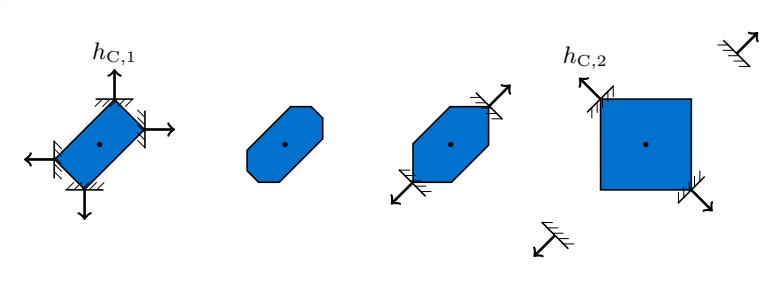

Figure A.4.: Polytopes according to the facet normal matrix in equation A.22 and the mapping matrix from equation A.23

Strategy. The above example illustrates that understanding improper $\mathcal{H}_{\mathrm{C}}$-representations is difficult, performed directly based on the associated $\mathbf{h}_{\mathrm{C}}$-space. On the other hand, the utilization of improper $\mathcal{H}_{\mathrm{C}}$-representations was straightforward in the scope of shape approximation since they are derived from and linked to a proper representation. As it will turn out, the $\mathbf{h}_{\mathrm{C}}$-space of an improper representation is always embedded in a corresponding, constructible, $\mathbf{h}_{\mathrm{E}}$-space of a proper $\mathcal{H}_{\mathrm{C}}$-representation. The relation between the original (improper) $\mathbf{h}_{\mathrm{C}}$-space and the constructed embedding (proper) $\mathbf{h}_{\mathrm{E}}$-space is exactly the same like between the approximated $\mathbf{h}_{\mathrm{A}}$-space and the original $\mathbf{h}_{\mathrm{C}}$-space for the dimensional reduction via shape approximation (see section 6.2). The constructed $\mathcal{H}_{\mathrm{C}}$-representation is denoted by *proper embedding* and directly transfers the analysis for proper representations. The required adopted notation for the proper embedding is analogue to the extended embedding, described on page 148 and summarized in column 4 of table 6.1 (page 121).

Proper Embedding. To construct a proper embedding, the rows $\mathbf{m}^{T}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h},\,i}$ of the group mapping matrix are separated in groups such that facets i and j of the same group:

1. are coherent (see page 41) and share the same facet validity cone ($\mathcal{C}_{\mathrm{FV}(i)} = \mathcal{C}_{\mathrm{FV}(j)}$) and
2. the rows of the group mapping matrix are equivalent ($\mathbf{m}^{T}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h},\,i} = \mathbf{m}^{T}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h},\,j}$).

The $\mathbf{h}_\mathrm{C}$-space can then be embedded in a new space of $\mathbf{h}_\mathrm{E}$-vectors, given by:

$$\mathbf{h}_\mathrm{E} = \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}\mathbf{h}_\mathrm{C} \tag{A.24}$$

$$\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}} = \begin{bmatrix} \mathbf{m}^{T}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h},\,i(1)} \\ \vdots \\ \mathbf{m}^{T}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h},\,i(n_\mathrm{G})} \end{bmatrix} \tag{A.25}$$

where $i(k)$ is any index to a facet of the k-th group of facets and n_G denotes the total number of groups. Given the space of $\mathbf{h}_\mathrm{E}$-vectors, the columns of the matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}$ span the linear subspace of the $\mathbf{h}_\mathrm{C}$-vectors. The resulting $\mathbf{h}_\mathrm{E}$-space belongs to a new $\mathcal{H}_\mathrm{C}$-representation which is proper, given the following discussion.

Based on the choice of the mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}$, every component of the $\mathbf{h}_\mathrm{E}$-vector directly represents facet distances for the same group of coherent facets. Hence, each row of the group mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{E}\mapsto\mathbf{h}}$ comprises a single one and zeros, otherwise. Additionally, each column of the group mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{E}\mapsto\mathbf{h}}$ contains only ones in rows for which the corresponding facets are coherent. In the scope of validity mapping (see subsection 3.2.3), this implies that setting the rows of coherent facets to zero, the rank of the adopted group mapping matrix $\mathbf{M}_{\mathbf{h}_\mathrm{E}\mapsto\mathbf{h}}$ becomes $n_\mathrm{C}-1$ and validity mapping is possible. In conclusion, the resulting $\mathcal{H}_\mathrm{C}$-representation is, indeed, proper and called the proper embedding.[3]

Example. Figure 6.8 on page 122 visualizes how the improper representation (shown in dark blue) is embedded in the $\mathbf{h}_\mathrm{E}$-space which is illustrated by the validity cone of the proper embedding in light blue. More details for this example are added in the following.

Forward Mapping. A vector $\mathbf{h}_\mathrm{C}$ can be lifted to the embedding $\mathbf{h}_\mathrm{E}$-space by equation A.24. The dark blue region in figure 6.8 is created based on this relation. As it might already be evident by this figure, the resulting $\mathbf{h}_\mathrm{E}$-vector is not necessarily valid since parts of the dark blue region are outside of the light blue validity cone. Therefore, an appropriate validity projection $\mathbf{P}_{\mathrm{V/E}}$ can be constructed in virtue of subsection 3.2.3 so that a valid $\mathbf{h}_\mathrm{E}$-vector is obtained by the mapping:

$$\mathbf{h}'_\mathrm{E} = \mathbf{P}_{\mathrm{V/E}}\left(\mathbf{h}_\mathrm{C}\right)\, \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}\, \mathbf{h}_\mathrm{C} \tag{A.26}$$

where the validity projection is denoted according to its dependency on the original $\mathbf{h}_\mathrm{C}$-vector.

[3] In fact, the proper embedding could be called an *orthogonal* representation with: $\mathbf{m}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h},\,i}\mathbf{m}^{T}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h},\,j} = 1$ for coherent facets and 0, otherwise. The principles of validity mapping are simplified in this case since the mapping directions $\mathbf{h}_{\mathrm{E},\mapsto}$ for a facet i are identical to $\mathbf{m}^{T}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h},\,i(i)}$.

Example (continued). Since different regions in $\mathbf{h}_C$-space cause different validity projections $\mathbf{P}_{V/E}(\mathbf{h}_C)$, the resulting valid $\mathbf{h}_E$-vectors literately fold the embedded $\mathbf{h}_C$-space to the validity cone. This situation is illustrated in figure A.5 for the simple example from figure 6.8. The part of the $\mathbf{h}_C$-space that is inside of the validity cone is not projected. The part of the $\mathbf{h}_C$-space that is behind the validity cone is projected to $\mathbf{0}$ for this example since the corresponding $\mathbf{h}_E$-vectors are outside of the non-empty polytope cone. The second dark blue part in front of the validity cone is folded onto the validity cone which is highlighted in blue. Finally, the white part in front of the validity cone is mapped to the front ray of the validity cone.

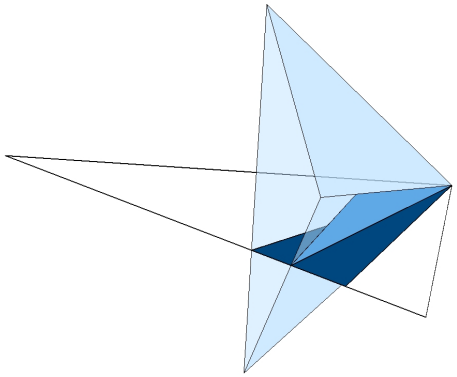

Figure A.5.: Validity cone of a proper embedding in light blue; the embedded $\mathbf{h}_C$-space is outlined by the wireframe triangle; regions of confined $\mathbf{h}_C$-vectors in dark blue and their corresponding valid $\mathbf{h}_E$-vectors in blue (if applicable)

Regions of $\mathbf{h}_C$-vectors. The observations above imply that the $\mathbf{h}_C$-space can be separated into regions for which different validity projections are applicable. According to subsection 3.2.3, the applicable region of the validity projection $\mathbf{P}_{V/E}(\mathbf{h}_C)$ is denoted by $\mathbf{A}_{\mathbf{P}_{V/E}(\mathbf{h}_C)}\ \mathbf{h}_E \leq \mathbf{0}$ so that the region in $\mathbf{h}_C$-space is given by:

$$\mathbf{A}_{\mathbf{P}_{V/E}(\mathbf{h}_C)}\ \mathbf{M}_{\mathbf{h}_C \mapsto \mathbf{h}_E}\ \mathbf{h}_C \leq \mathbf{0}. \tag{A.27}$$

Since only a finite number of validity projections exists, the $\mathbf{h}_C$-space is also separated into a finite number of such regions for which specific validity projections apply.

Conclusion. Given the proper embedding and particularly the mapping from equation A.26, the a-type for a given polytope $P(\mathbf{h}_C)$ can be quickly identified and measure computations can be performed, as usual, based on the proper embedding. The overall procedure can be further shortened by transferring the operations on the $\mathbf{h}_E$-vectors to the $\mathbf{h}_C$-space. This step, however, is straightforward and does not provide a significantly better understanding of the $\mathbf{h}_C$-space. Since it is also extensive in its presentation, it is explained in appendix A.2 and yields to a concept of measure computation that is fully analogue to the one for proper $\mathcal{H}_C$-representations (unified partitions, measure polynomials). On the other hand, the mapping matrix $\mathbf{P}_{V/E}(\mathbf{h}_C)\ \mathbf{M}_{\mathbf{h}_C \mapsto \mathbf{h}_E}$ from $\mathbf{h}_C$-vectors to valid $\mathbf{h}_E$-vectors is central for the discussion of the main difference between proper and

improper representations, given in the next subsection. It is, therefore, already evident that the proper embedding is the key to the analysis of improper representations.

A.2.2 Confinement

Motivation. This subsection clarifies the main difference between proper and improper $\mathcal{H}_\mathrm{C}$-representations which constitutes the inherent complexity of improper representations. It originates from two arguments that were used to motivate validity and ill-conditioned representations.

1. Two distinct $\mathbf{h}_\mathrm{C}$-vectors must represent different polytopes $P(\mathbf{h}_\mathrm{C})$. Otherwise, one of these $\mathbf{h}_\mathrm{C}$-vectors is redundant and should be excluded from being valid.

2. For an $\mathbf{h}$-vector, it must hold $h_i = h(P(\mathbf{h}), \mathbf{a}_i^T)$ so that a facet immediately reappears when the dynamic evolution based on a growth rate $\mathbf{g}(t) = \frac{\mathrm{d}\mathbf{h}}{\mathrm{d}t}$ indicates such a situation.

While the second argument is used to define the validity of $\mathbf{h}$- and $\mathbf{h}_\mathrm{C}$-vectors, the first argument appears only implicitly throughout section 3.2. This was possible since both arguments are redundant for proper representations. For improper representations, this is not the case anymore. The following two paragraphs clarify this situation while the concept of validity is expanded in subsequent paragraphs by a new property of $\mathbf{h}_\mathrm{C}$-vectors to address the difference of the above two arguments. This final theoretical analysis discloses a possible dilemma in the context of crystal shape approximations.

Redundancy for proper representations. First, it is shown that the validity of two distinct vectors $\mathbf{h}_\mathrm{C}$ and $\mathbf{h}'_\mathrm{C}$ implies that the corresponding polytopes cannot be equal: $P(\mathbf{h}_\mathrm{C}) \neq P(\mathbf{h}'_\mathrm{C})$. Given two valid and distinct $\mathbf{h}_\mathrm{C}$-vectors, their support functions are different by the definition of validity. Since differing support functions imply that the associated polytopes $P(\mathbf{h}_\mathrm{C})$ have a distinct shape or size, argument one holds. This is true for any $\mathcal{H}_\mathrm{C}$-representation. Secondly, it is shown that once a $\mathbf{h}_\mathrm{C}$-vector is invalid, more vectors $\mathbf{h}'_\mathrm{C}$ exist so that $P(\mathbf{h}_\mathrm{C}) = P(\mathbf{h}'_\mathrm{C})$ holds. Given an invalid $\mathbf{h}_\mathrm{C}$-vector in the context of a proper $\mathcal{H}_\mathrm{C}$-representation, there exists a direction in $\mathbf{h}_\mathrm{C}$-space which can be used to change the $\mathbf{h}_\mathrm{C}$-vector without changing the shape of the corresponding polytope. In conclusion, for proper representations, arguments one and two for validity are indeed redundant.

Irredundancy for improper representations. If a representation is improper, validity mapping is not always possible because no direction for validity mapping exists. Hence, an invalid $\mathbf{h}_\mathrm{C}$-vector might exist so that the polytope $P(\mathbf{h}_\mathrm{C})$ cannot be expressed by a different and possibly valid vector $\mathbf{h}'_\mathrm{C}$ (see the right of figure A.4 and the discussion on page 167). In other words, shifting the invalid vector $\mathbf{h}_\mathrm{C}$ always changes the corresponding polytope so that arguments one and two for validity are no longer equivalent.

Confined $\mathbf{h}_\mathrm{C}$-vectors. It can now be concluded that the two arguments that were used to motivate the validity of $\mathbf{h}_\mathrm{C}$-vectors for proper representations are, in fact, distinct properties of a $\mathbf{h}_\mathrm{C}$-vector. While the validity that reflects argument 2 from above is already defined in subsection 3.2.2, the new classification of $\mathbf{h}_\mathrm{C}$-vectors will be called confinement and reflects argument 1.

A $\mathbf{h}_\mathrm{C}$-vector is called confined when for any direction $\mathbf{u} \in S^{(n_\mathrm{C}-1)}$ and any positive scalar $\varepsilon > 0$ it holds: $P(\mathbf{h}_\mathrm{C}) \neq P(\mathbf{h}_\mathrm{C} + \varepsilon\mathbf{u})$ and $P(\mathbf{h}_\mathrm{C}) \neq P(\mathbf{h}_\mathrm{C} - \varepsilon\mathbf{u})$. The $\mathbf{h}_\mathrm{C}$-vector

cannot be changed without affecting the shape $P(\mathbf{h}_C)$. This describes a not necessarily compact set of $\mathbf{h}_C$-vectors. To ensure that also limiting cases are contained in the set of confined $\mathbf{h}_C$-vectors, a $\mathbf{h}_C$ vector is also confined when a (convergent) series of confined vectors $\mathbf{h}'_{C,i}$ exists for which:

$$\mathbf{h}_C = \lim_{i \to \infty} \mathbf{h}_{C,i} \tag{A.28}$$

holds. Unconfined $\mathbf{h}_C$-vectors are typically identified when a direction $\mathbf{u} \in S^{(n_C-1)}$ and fixed positive scalar $\varepsilon > 0$ exists so that it holds $P(\mathbf{h}_C) = P(\mathbf{h}_C + \varepsilon\mathbf{u}) = P(\mathbf{h}_C - \varepsilon\mathbf{u})$. This is, however, not necessarily applicable for $\mathbf{h}_C$-vectors at the boundary of the set of confined $\mathbf{h}_C$-vectors. Confinement can only be defined for $\mathbf{h}_C$-vectors and does not exist for $\mathbf{h}$-vectors. The set of confined $\mathbf{h}_C$-vectors is not convex which is explained in subsection A.2.3.

Determining confinement. The definition of confinement resembles argument 1 for validity but is not suited to compute whether a given $\mathbf{h}_C$-vector is confined or unconfined. A corresponding equation is, therefore, provided in the following based on the proper embedding and the corresponding $\mathbf{h}_E$-vector: $\mathbf{h}_E = \mathbf{M}_{\mathbf{h}_C \mapsto \mathbf{h}_E} \mathbf{h}_C$. For this vector, a validity projection $\mathbf{P}_{V/E}(\mathbf{h}_C)$ exists according to subsection 3.2.3 so that the vector $\mathbf{P}_{V/E}(\mathbf{h}_C)\ \mathbf{h}_E$ is valid. This validity projection also has a defined region in which it is applicable, given by equation A.27. Given that the vector $\mathbf{h}_C$ resides in the interior of this region, the condition:

$$\operatorname{rank}\left(\mathbf{P}_{V/E}(\mathbf{h}_C)\ \mathbf{M}_{\mathbf{h}_C \mapsto \mathbf{h}_E}\right) = n_C, \tag{A.29}$$

implies that a region of vectors $\mathbf{h}'_C$ with $\|\mathbf{h}'_C - \mathbf{h}_C\| \leq \varepsilon$ and $\varepsilon > 0$ exists for which all vectors:

$$\mathbf{h}'_E = \mathbf{P}_{V/E}(\mathbf{h}_C)\ \mathbf{M}_{\mathbf{h}_C \mapsto \mathbf{h}_E} \mathbf{h}'_C \tag{A.30}$$

are different and valid. The vector $\mathbf{h}_C$ would, hence, be confined. While confinement cannot be concluded directly for $\mathbf{h}_C$ residing on the boundary of the region, based on the equation A.29, this is not required. The region of confined $\mathbf{h}_C$-vectors is closed according to the definition of confined $\mathbf{h}_C$-vectors, so that equation A.29 also indicates confinement when the vector $\mathbf{h}_C$ is at the boundary of the applicable region[4]. In summary, the fulfillment of equation A.29 directly ensures the confinement of a $\mathbf{h}_C$-vector and all $\mathbf{h}_C$-vectors in the region defined by equation A.27.

Simplified determination of confinement. Using equation A.29 to determine the confinement of a $\mathbf{h}_C$-vector requires to construct the specific validity mapping $\mathbf{P}_{V/E}(\mathbf{h}_C)$. Therefore, the invalid facets must be determined and facet specific validity mappings must be selected. The following derivation demonstrates that the latter step is not required to determine confinement. The basic idea is to utilize that the null space of the projection $\mathbf{P}_{V/E}(\mathbf{h}_C)$ is only dependent on the set of invalid facets.

Based on the rank-nullity theorem, the matrix $\mathbf{P}_{V/E}(\mathbf{h}_C)\ \mathbf{M}_{\mathbf{h}_C \mapsto \mathbf{h}_E}$ can only have the full column rank n_C when the dimension of the null space is zero. This means that a full rank n_C is only obtained when no vector $\mathbf{h}'_C \neq \mathbf{0}$ exists with [Lorenz, 2003]:

$$\mathbf{P}_{V/E}(\mathbf{h}_C)\ \mathbf{M}_{\mathbf{h}_C \mapsto \mathbf{h}_E} \mathbf{h}'_C = \mathbf{0}. \tag{A.31}$$

[4]Note that at the boundary of an applicable region, two or more validity projection matrices $\mathbf{P}_{V/E}$ are valid. An implementation of the above approach must ensure that the validity projection matrix $\mathbf{P}_{V/E}(\mathbf{h}_C)$ with the smallest possible null space is constructed to determine confinement.

If a vector $\mathbf{h}'_{\mathrm{C}} \neq \mathbf{0}$ exists that fulfills this equation, the vector $\mathbf{h}_{\mathrm{C}}$ is unconfined. In such a case, the vector $\mathbf{h}'_{\mathrm{C}}$ indicates a linear combination of columns of the matrix $\mathbf{M}_{\mathbf{h}_{\mathrm{C}} \mapsto \mathbf{h}_{\mathrm{E}}}$ so that the vector $\mathbf{M}_{\mathbf{h}_{\mathrm{C}} \mapsto \mathbf{h}_{\mathrm{E}}} \mathbf{h}'_{\mathrm{C}}$ is an element of the null space of the projection $\mathbf{P}_{\mathrm{V/E}}(\mathbf{h}_{\mathrm{C}})$. Hence, a vector $\mathbf{h}'_{\mathrm{C}}$ that fulfills equation A.31 exists if and only if the columns of the matrix $\mathbf{M}_{\mathbf{h}_{\mathrm{C}} \mapsto \mathbf{h}_{\mathrm{E}}}$ and an arbitrarily chosen set of base vectors for the null space of the matrix $\mathbf{P}_{\mathrm{V/E}}(\mathbf{h}_{\mathrm{C}})$ are linearly dependent. Given the principles of the validity mapping for proper representations (given in subsection 3.2.3), the null space of the projection $\mathbf{P}_{\mathrm{V/E}}(\mathbf{h}_{\mathrm{C}})$ is spanned by the mapping directions $\mathbf{h}_{\mathrm{E},\mapsto,i}$ which depend, individually, only on the disappeared facet group. Hence, the confinement condition from equation A.29 can be reformulated. A vector $\mathbf{h}_{\mathrm{C}}$ is confined if it holds:

$$\operatorname{rank}\left[\mathbf{h}_{\mathrm{E},\mapsto,1} \quad \ldots \quad \mathbf{h}_{\mathrm{E},\mapsto,n_{\mathrm{D}}} \quad \mathbf{M}_{\mathbf{h}_{\mathrm{C}} \mapsto \mathbf{h}_{\mathrm{E}}}\right] = n_{\mathrm{D}} + n_{\mathrm{C}} \tag{A.32}$$

where the vectors $\mathbf{h}_{\mathrm{E},\mapsto,i}$ are the appropriate mapping directions for the validity correction (given in subsection 3.2.3) of the vector $\mathbf{M}_{\mathbf{h}_{\mathrm{C}} \mapsto \mathbf{h}_{\mathrm{E}}} \mathbf{h}_{\mathrm{C}}$. Finding the required mapping directions $\mathbf{h}_{\mathrm{E},\mapsto,i}$ is only based on the facet validity cones $\mathcal{C}_{\mathrm{FV}(i)}$ and, hence, much simpler than constructing the overall validity mapping.

Summary and Conclusions. This subsection has shown that, in general, the validity of a $\mathbf{h}_{\mathrm{C}}$-vector (support values $h(P(\mathbf{h}_{\mathrm{C}}), \mathbf{a}_i)$ match the modeled facet distance $\mathbf{h}_i = \mathbf{M}_{\mathbf{h}_{\mathrm{C}} \mapsto \mathbf{h}} \mathbf{h}_{\mathrm{C}}$) is a different property of a $\mathbf{h}_{\mathrm{C}}$-vector than confinement (changing the vector $\mathbf{h}_{\mathrm{C}}$ also changes the shape $P(\mathbf{h}_{\mathrm{C}})$), even though both properties are similar and related. For these properties, the following relations hold.

- A valid $\mathbf{h}_{\mathrm{C}}$-vector is necessarily confined.
- An invalid $\mathbf{h}_{\mathrm{C}}$-vectors might be confined or unconfined.
- Unconfined vectors are invalid.
- Confined vectors can be valid or invalid.
- For proper representations, the terms validity and confinement are equivalent.

A particularly problematic relation is that an invalid $\mathbf{h}_{\mathrm{C}}$-vector might be confined which immediately implies that validity cannot be restored without changing the corresponding particle shape. This is not surprising since proper representations are defined in subsection 3.2.4 such that they ensure that validity mapping is always applicable. In general, equation A.32 implies that $n_{\mathrm{E}} - n_{\mathrm{C}}$ facets can grow out before a $\mathbf{h}_{\mathrm{C}}$-vector becomes unconfined. While unconfined $\mathbf{h}_{\mathrm{C}}$-vectors can be mapped to corresponding confined $\mathbf{h}_{\mathrm{C}}$-vectors, analogue to validity mapping, the derivation is given in appendix A.2 since no case studies are presented based on these derivations.

Conclusions for shape approximation. When the confinement condition from equation A.32 is viewed in the scope of shape approximations (see section 6.2), it implies that, typically, $n_{\mathrm{C}} - n_{\mathrm{A}}$ facets can grow out before a $\mathbf{h}_{\mathrm{C}}$-vector becomes unconfined. This means that the creation of the first invalid facet can, typically, not be corrected. However, it also demonstrates that a shape approximation can be constructed such that the creation of a specific invalid facet implies unconfinement so that validity can be maintained. For such a case, the mapping direction $\mathbf{h}_{\mathrm{C},\mapsto}$ of the selected facet must be contained in the

range of the mapping $\mathbf{M}_{\mathbf{h}_A \mapsto \mathbf{h}_C}$. This approach contains a contradiction. A shape approximation should typically yield a good approximation of the particle volume. On the other hand, a facet that disappears and reappears at the crystal surface, possibly more than once (see Snyder et al. [2007]), does not have a large impact on the crystal volume because it truncates only small parts of the crystal (see the discussion of the urea case study on pages 6.2.1 and 6.2.2 or the paracetamol case study on page 6.3.3). It follows that the selection of the linear subspace for an approximation can either be in favor of a good approximation of the volume measure, or in favor of modeling the appearance and disappearance of a specific facet.

A.2.3 Confinement mapping

Motivation. In subsection A.2.2, confinement is introduced as a property of $\mathbf{h}_C$-vectors that must be distinguished from validity for improper representations. While a method to determine the confinement of a $\mathbf{h}_C$-vector is provided, no procedure is given to correct unconfined $\mathbf{h}_C$-vectors. Additionally, it is not discussed how invalid vectors could be corrected. This subsection supplements these details for improper representations.

Example. Even though confinement is strongly linked to validity, it involves several peculiarities that are outlined in the following example. As an example, an improper representation is defined by the matrices:

$$\mathbf{A} = \begin{bmatrix} 1 & -1 & 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ 0 & 0 & 1 & -1 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}^T \tag{A.33}$$

$$\mathbf{M}_{\mathbf{h}_C \mapsto \mathbf{h}} = \begin{bmatrix} 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 \\ 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \end{bmatrix}^T. \tag{A.34}$$

Shapes of this representation are drawn in figure A.7 and the validity cone of the proper embedding is drawn in light blue in figure A.6. Similar to figure A.5, confined regions of $\mathbf{h}_C$-vectors are highlighted in dark blue. Two regions of confined $\mathbf{h}_C$-vectors are visible so that the complete region of confined $\mathbf{h}_C$-vectors is not convex. In contrast to that, the region of valid $\mathbf{h}_C$-vectors is always convex. This comes along with a second peculiarity. All $\mathbf{h}_C$-vectors between the two confined regions are mapped to the right ray of the validity cone. Since this holds for the two inner rays of the confined regions, two equally applicable confined $\mathbf{h}_C$-vectors exist that represent the same polytope. The same situation can be retraced based on figure A.7. The situation in the middle can either be represented by the confined representation on its left, or by the confined representation on its right. This illustrates that unconfined $\mathbf{h}_C$-vectors do not necessarily have a unique corresponding confined $\mathbf{h}_C$-vector. Invalid $\mathbf{h}_C$-vectors always have a unique corresponding valid $\mathbf{h}_C$-vector.

Regions of $\mathbf{h}_E$-vectors. The algorithm for confinement mapping requires two components that are introduced in this and the following paragraph. In this paragraph, the region of $\mathbf{h}_E$-vectors is discussed that is covered by a corresponding region of $\mathbf{h}_C$-vectors. The regions of $\mathbf{h}_E$-vectors are drawn in blue in figures A.5 and A.6 while the regions of $\mathbf{h}_C$-vectors are drawn in dark blue. Between both regions, the following mapping exists:

$$\mathbf{h}_C = \left(\mathbf{P}_{V/E}\left(\mathbf{h}_C\right) \mathbf{M}_{\mathbf{h}_C \mapsto \mathbf{h}_E}\right)^+ \mathbf{h}_E \tag{A.35}$$

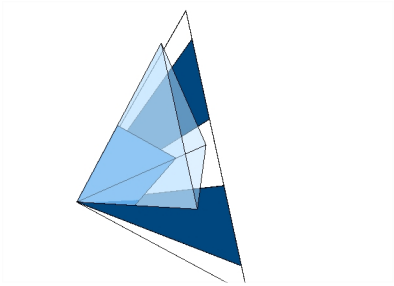

Figure A.6.: Validity cone of a proper embedding in light blue, confined regions of the embedded improper representation shown in dark blue and the corresponding projections onto the validity cone shown in blue (if applicable)

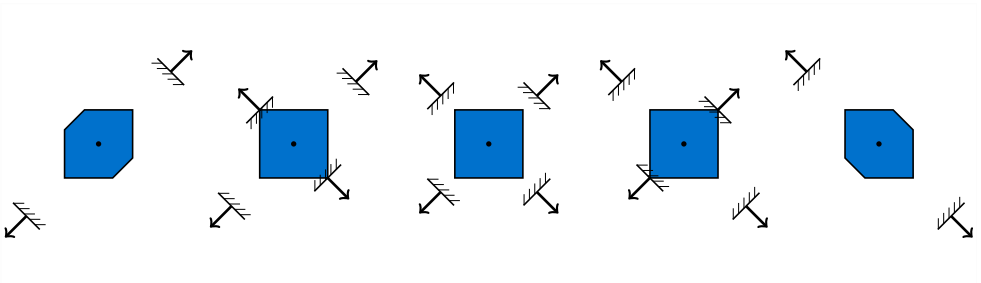

Figure A.7.: Polytopes according to the facet normal matrix in equation A.33 and mapping matrix from equation A.34

which creates the original $\mathbf{h}_\mathrm{C}$-vector according to equation A.26 when the matrix $\mathbf{P}_\mathrm{V/E}\left(\mathbf{h}_\mathrm{C}\right)$ $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}$ has full column rank.

Equation A.27 describes the dark blue colored regions in $\mathbf{h}_\mathrm{C}$-space. If the vector $\mathbf{h}_\mathrm{C}$ is substituted by equation A.35, the blue regions of the $\mathbf{h}_\mathrm{E}$-space are covered. But the region that is then described by:

$$\mathbf{A}_{\mathbf{P}_\mathrm{V/E}\left(\mathbf{h}_\mathrm{C}\right)\,\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}}\left(\mathbf{P}_\mathrm{V/E}\left(\mathbf{h}_\mathrm{C}\right)\,\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}\right)^{+}\mathbf{h}_\mathrm{E}\leq\mathbf{0} \tag{A.36}$$

also contains the null space of the matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}\left(\mathbf{P}_\mathrm{V/E}\left(\mathbf{h}_\mathrm{C}\right)\,\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}\right)^{+}$. Not yet considered is that the region of $\mathbf{h}_\mathrm{E}$-vectors must be limited to the range of the validity projection $\mathbf{P}_\mathrm{V/E}$ which is indicated by the blue regions. The aforementioned null space is the orthogonal complement of this range so that eliminating this null space fixes $\mathbf{h}_\mathrm{E}$-vectors simultaneously to the range of the validity projection $\mathbf{P}_\mathrm{V/E}$ (the blue regions).[5] The orthogonal complement for the range of the validity projection $\mathbf{P}_\mathrm{V/E}$ can be constructed

[5]The following justifies this statement. The matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}$ does not contribute to the null space. It has a full column rank, otherwise the representation would be ill-conditioned. Then it holds that the null space of a pseudo inverse $\mathbf{M}^{+}$ equals the null space of the matrix $\mathbf{M}^{T}$ so that the null space of $\left(\mathbf{P}_\mathrm{V/E}\left(\mathbf{h}_\mathrm{C}\right)\,\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}\right)^{+}$ equals the null space of $\left(\mathbf{P}_\mathrm{V/E}\left(\mathbf{h}_\mathrm{C}\right)\,\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}\right)^{T}$. Additionally, the null space of a matrix $\mathbf{M}$ equals the orthogonal complement of the range of its transpose $\mathbf{M}^{T}$ so that the null space of $\left(\mathbf{P}_\mathrm{V/E}\left(\mathbf{h}_\mathrm{C}\right)\,\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}\right)^{T}$ equals the orthogonal complement of the range for $\mathbf{P}_\mathrm{V/E}\left(\mathbf{h}_\mathrm{C}\right)\,\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}$. Again, the matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}$ does not contribute to the range. Hence, it can be concluded that the orthogonal complement of the range of the validity projection $\mathbf{P}_\mathrm{V/E}$ indeed represents the null space of the matrix $\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}\left(\mathbf{P}_\mathrm{V/E}\left(\mathbf{h}_\mathrm{C}\right)\,\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}\right)^{+}$.

and the corresponding base vectors are stored as rows in a matrix $\mathbf{A}_\perp$. Demanding $\mathbf{A}_\perp \mathbf{h}_\mathrm{E} = \mathbf{0}$ finally fixes the $\mathbf{h}_\mathrm{E}$-vecors to the blue regions. This region of $\mathbf{h}_\mathrm{E}$-vectors is then provided as a $\mathcal{H}$-representation by:

$$\begin{bmatrix} \mathbf{A}_\perp \\ -\mathbf{A}_\perp \\ \mathbf{A}_{\mathbf{P}_\mathrm{V/E}(\mathbf{h}_\mathrm{C})\ \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}} \left(\mathbf{P}_\mathrm{V/E}(\mathbf{h}_\mathrm{C})\ \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}\right)^+ \end{bmatrix} \mathbf{h}_\mathrm{E} \leq \mathbf{0}. \tag{A.37}$$

Existence of confined vectors. The above paragraph constitutes one required component for the correction of unconfined vectors. A second component considers that a confined vector $\mathbf{h}'_\mathrm{C}$ indeed exists for each unconfined $\mathbf{h}_\mathrm{C}$-vector. This is not obvious for confinement as it is evident for the validity of $\mathbf{h}_\mathrm{C}$-vectors.

Given is an unconfined vector $\mathbf{h}_\mathrm{C}$. If this vector represents an empty polytope, $\mathbf{h}'_\mathrm{C} = \mathbf{0}$ can be chosen which is always a confined vector.[6] If this is not the case, it exists a direction $\mathbf{h}_{\mathrm{C},\mapsto}$ in $\mathbf{h}_\mathrm{C}$-space along which the given vector can be moved freely (see equation A.31). It affects one or more facet groups that do not touch the polytope $P(\mathbf{h}_\mathrm{C})$. One of these facet groups is selected. Now, a finite scalar γ and the corresponding vector $\mathbf{h}'_\mathrm{C}(\gamma) = \mathbf{h}_\mathrm{C} + \gamma\mathbf{h}_{\mathrm{C},\mapsto}$ exist so that a facet of that group does touch the polytope $P(\mathbf{h}_\mathrm{C})$. However, the polytope $P(\mathbf{h}'_\mathrm{C})$ might be different from $P(\mathbf{h}_\mathrm{C})$ when the wrong facet group is selected. In this case, a different facet group touches the polytope $P(\mathbf{h}_\mathrm{C})$ at some vector $\mathbf{h}'_\mathrm{C}(\gamma')$ with $|\gamma'| \leq |\gamma|$. If this is the case, γ' is assumed instead of γ for the computation of the new vector $\mathbf{h}'_\mathrm{C}$. This procedure is repeated until a vector $\mathbf{h}'_\mathrm{C}$ is found for which $P(\mathbf{h}'_\mathrm{C}) = P(\mathbf{h}_\mathrm{C})$ holds and a facet group now touches the polytope $P(\mathbf{h}_\mathrm{C})$ that did not touch the polytope for the vector $\mathbf{h}_\mathrm{C}$. Since a new facet group touches the polytope $P(\mathbf{h}_\mathrm{C})$, a new validity projection $\mathbf{P}_\mathrm{V/E}(\mathbf{h}'_\mathrm{C})$ is applicable while the null space of the matrix $\mathbf{P}_\mathrm{V/E}(\mathbf{h}'_\mathrm{C})\ \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}$ is reduced in its dimension. At this point, either the null space of the matrix $\mathbf{P}_\mathrm{V/E}(\mathbf{h}'_\mathrm{C})\ \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}$ is already empty and indicates confinement for the vector $\mathbf{h}'_\mathrm{C}$ or the procedure can be repeated until this is the case. This concludes that a confined vector $\mathbf{h}'_\mathrm{C}$ exists for each unconfined vector $\mathbf{h}_\mathrm{C}$ while $P(\mathbf{h}'_\mathrm{C}) = P(\mathbf{h}_\mathrm{C})$ holds.

Confinement mapping. Based on the two preceding paragraphs, it is now possible to provide a procedure to find a confined vector $\mathbf{h}'_\mathrm{C}$ that represents the same polytope as a given unconfined vector $\mathbf{h}_\mathrm{C}$. First, the unconfined vector $\mathbf{h}_\mathrm{C}$ is lifted to the $\mathbf{h}_\mathrm{E}$-space via: $\mathbf{h}_\mathrm{E} = \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}\mathbf{h}_\mathrm{C}$. Secondly, validity mapping is applied to obtain the valid vector $\mathbf{h}'_\mathrm{E}$. In the next step, the regions of $\mathbf{h}_\mathrm{E}$-vectors are considered that are covered by the different regions of $\mathbf{h}_\mathrm{C}$-vectors according to equation A.37. Once a corresponding region is found, the mapping from equation A.35 is applied to obtain a confined vector $\mathbf{h}'_\mathrm{C}$.

Two remarks are necessary. First, it might not be clear whether a region according to equation A.37 can be found. However, this region must exist since a corresponding confined $\mathbf{h}_\mathrm{C}$-vector exists which is shown in the preceding paragraph. Secondly, it is possible that more than one region is identified according to equation A.37 which indicates that the vector $\mathbf{h}'_\mathrm{C}$ is not necessarily uniquely determined (see the example from the beginning of this subsection).

[6] The vector $\mathbf{h}_\mathrm{C} = \mathbf{0}$ represents the polytope $P = \{\mathbf{0}\}$ for any $\mathcal{H}_\mathrm{C}$-representation. Any direction $\mathbf{h}_{\mathrm{C},\mapsto}$ affects at least one facet so that either $P(\mathbf{h}_\mathrm{C} + \varepsilon\mathbf{h}_{\mathrm{C},\mapsto})$ does not contain $\mathbf{0}$ or $P(\mathbf{h}_\mathrm{C} - \varepsilon\mathbf{h}_{\mathrm{C},\mapsto})$ does not contain $\mathbf{0}$ holds for some $\varepsilon > 0$. The position of the polytope is fixed since ill-conditioned representations are not discussed so that either $P(\mathbf{h}_\mathrm{C} + \varepsilon\mathbf{h}_{\mathrm{C},\mapsto}) = \emptyset$ or $P(\mathbf{h}_\mathrm{C} - \varepsilon\mathbf{h}_{\mathrm{C},\mapsto}) = \emptyset$ holds and $\mathbf{h}_\mathrm{C} = \mathbf{0}$ is confined.

Validity mapping. Finding a valid vector $\mathbf{h}'_{\mathrm{C}}$ for a given invalid vector $\mathbf{h}_{\mathrm{C}}$ while the shape is retained ($P(\mathbf{h}'_{\mathrm{C}}) = P(\mathbf{h}_{\mathrm{C}})$) is usually not possible for improper representations. The validity cone might even be empty in such cases. The closest solution to this problem is to apply confinement mapping while equation A.32 indicates how the improper representation must be constructed so that the validity of a specific facet can be restored. If confinement mapping cannot restore the validity of the $\mathbf{h}_{\mathrm{C}}$-vector, it is necessary to solve an optimization problem to preserve shape characteristics as well as possible. One possible strategy is to minimize the distance between the invalid vector $\mathbf{h}_{\mathrm{C}}$ and the valid vector $\mathbf{h}'_{\mathrm{C}}$, resulting in a linear objective function with linear inequality constraints:

$$\mathbf{h}'_{\mathrm{C}} = \underset{\mathbf{h}'_{\mathrm{C}} \in \mathcal{C}_{\mathrm{V}}}{\operatorname{argmin}} \left\| \mathbf{h}_{\mathrm{C}} - \mathbf{h}'_{\mathrm{C}} \right\|. \tag{A.38}$$

This optimization is equivalent to the orthogonal projection of the vector $\mathbf{h}_{\mathrm{C}}$ to the validity cone $\mathcal{C}_{\mathrm{V}}$.

A.2.4 Decomposition

Motivation. Section 3.3 and the previous subsections already provide the basic principles of measure computation, confinement and validity for improper representations. However, the required information is not yet generated in a systematic manner to allow an efficient application. This subsection describes a decomposition algorithm that is similar to the decomposition of the validity cone for proper representations. This decomposition allows the accumulation of any information *a priori* to applications of this data. It is assumed that every information of the proper embedding is computed according to chapter 3 and available.

Decomposed domain. For proper representations, all unified partitions are obtained by the decomposition of the validity cone $\mathcal{C}_{\mathrm{V}}$. This domain is not a reasonable starting point for improper representations since the set of represented polytopes is given by the set of confined vectors and not by the set of valid vectors. An additional problem is raised by not even knowing the region of confined $\mathbf{h}_{\mathrm{C}}$-vectors, yet. Recall from the previous subsection that this region is not necessarily convex. However, since $\mathbf{h}_{\mathrm{C}}$-vectors for empty polytopes are always unconfined, the cone of non-empty polytopes $\mathcal{C}_{\mathrm{FV}(0)}$ is chosen as the starting domain for the decomposition. It is obtained from the proper embedding by:

$$\mathbf{A}_{\mathrm{FV}(0)} = \mathbf{A}_{\mathrm{FV}(0)/\mathrm{E}} \mathbf{M}_{\mathbf{h}_{\mathrm{C}} \mapsto \mathbf{h}_{\mathrm{E}}}. \tag{A.39}$$

Construction of unified partitions. To retain the terminology of proper representations, the regions in which a set of measure coefficients $c^{\mathrm{measure}}_{p,(i,j,k)}$ remains valid is also called a unified partition. However, vectors of a unified partition are not necessarily valid so that they do not represent a maximum a-type. The following steps are used to compute the region of $\mathbf{h}_{\mathrm{C}}$-vectors for which a set of measure coefficients $c^{\mathrm{measure}}_{p,(i,j,k)}$ remains valid. It additionally describes the information that is gathered during the decomposition. It starts with an arbitrary vector $\mathbf{h}_{\mathrm{C}}$.

1. The vector $\mathbf{h}_{\mathrm{C}}$ is lifted to the proper embedding by: $\mathbf{h}_{\mathrm{E}} = \mathbf{M}_{\mathbf{h}_{\mathrm{C}} \mapsto \mathbf{h}_{\mathrm{E}}} \mathbf{h}_{\mathrm{C}}$.
2. The validity projection $\mathbf{P}_{\mathrm{V/E}}(\mathbf{h}_{\mathrm{E}})$ is selected while the matrix of facet normals $\mathbf{A}_{\mathbf{P}_{\mathrm{V/E}}}(\mathbf{h}_{\mathrm{E}})$ indicates the applicable region for this projection. The corresponding valid $\mathbf{h}_{\mathrm{E}}$-vector is calculated by: $\mathbf{h}'_{\mathrm{E}} = \mathbf{P}_{\mathrm{V/E}}(\mathbf{h}_{\mathrm{E}})\, \mathbf{h}_{\mathrm{E}}$. If the vector $\mathbf{h}_{\mathrm{E}}$ is already

valid, the resulting projection matrix is $\mathbf{P}_{\mathrm{V/E}}(\mathbf{h}_\mathrm{E}) = \mathbf{I}_{n_\mathrm{E}}$, the matrix of normal vectors $\mathbf{A}_{\mathbf{P}_{\mathrm{V/E}}}(\mathbf{h}_\mathrm{E})$ is empty and $\mathbf{h}'_\mathrm{E} = \mathbf{h}_\mathrm{E}$ holds.

3. The confinement of the vector $\mathbf{h}_\mathrm{C}$ is determined based on equation A.29.

4. Based on the valid vector $\mathbf{h}'_\mathrm{E}$, the matching unified partition q of the proper embedding is determined so that $\mathbf{A}_{\mathrm{U}(q)/\mathrm{E}}\mathbf{h}'_\mathrm{E} \leq \mathbf{0}$ holds.

5. The matrix:
$$\mathbf{A}_{\mathrm{U},p} = \begin{bmatrix} \mathbf{A}_{\mathbf{P}_{\mathrm{V/E}}}(\mathbf{h}_\mathrm{E}) \\ \mathbf{A}_{\mathrm{U},q/\mathrm{E}}\,\mathbf{P}_{\mathrm{V/E}}(\mathbf{h}_\mathrm{E}) \end{bmatrix} \mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}} \tag{A.40}$$
is accumulated so that:
$$\mathbf{A}_{\mathrm{U},p}\mathbf{h}_\mathrm{C} \leq \mathbf{0} \tag{A.41}$$
describes all $\mathbf{h}_\mathrm{C}$-vectors for which the validity mapping is applicable and the selected unified partition is valid.

6. The region of $\mathbf{h}_\mathrm{E}$-vectors that is covered by this unified partition is stored as a $\mathcal{H}$-representation according to equation A.37.

Given the coefficients $c^{\mathrm{measure}}_{q,(i,j,k)/\mathrm{E}}$ for the unified partition q and the mapping matrix $\mathbf{P}_{\mathrm{V/E}}(\mathbf{h}_\mathrm{E})\,\mathbf{M}_{\mathbf{h}_\mathrm{C}\mapsto\mathbf{h}_\mathrm{E}}$, the corresponding coefficients $c^{\mathrm{measure}}_{q,(i,j,k)}$ for the improper representation can be calculated.

Decomposition algorithm. To identify all required unified partitions that cover the set of all confined $\mathbf{h}_\mathrm{C}$-vectors, the following decomposition is performed. This algorithm uses the same principles that are applied for the decomposition of the validity cone for proper representations (see section 3.3). The domain to be decomposed is initialized with the non-empty polytope cone: $\mathcal{D} = \mathcal{C}_{\mathrm{FV}(0)}$, given by equation A.39. Since this domain does not necessarily remain convex throughout the algorithm, it is provided as a partitioning into convex regions $\mathcal{D}_i$: $\mathcal{D} = \cup_i \mathcal{D}_i$. After this initialization, the algorithm performs the following steps.

1. **Generate random vector** $\mathbf{h}_\mathrm{C} \in \mathcal{D}_1$
   The natively scaled extreme rays of $\mathcal{D}_1$ are multiplied by a random number and added to obtain $\mathbf{h}_\mathrm{C}$. The random numbers are chosen to be sufficiently large to guarantee that $\mathbf{h}_\mathrm{C}$ is not at the boundary of $\mathcal{D}_1$. If $\mathbf{h}_\mathrm{C}$ would be at the boundary of $\mathcal{D}_1$, it is likely that it is also on the boundary of some unified partition.

2. **Generate partition**
   A partition according equation A.40 and A.41 is generated and denoted by $\mathcal{C}_\ominus$. If the obtained region is not n_C-dimensional, it is returned to step 1. This event is unlikely so that no infinite loop is created.

3. **Accept new partition**
   If the vector $\mathbf{h}_\mathrm{C}$ is confined, the region is stored as a unified partition. If the vector $\mathbf{h}_\mathrm{C}$ is unconfined, the region is not stored. In either case, the remaining domain is updated by the set difference $\mathcal{D}\backslash\mathcal{C}_\ominus$ and stored via its partitioning into cones: $\mathcal{D} = \cup_i \mathcal{D}_i$.

4. **Termination check**
   If the domain $\mathcal{D}$ is empty, the decomposition is complete and the algorithm terminates. If the domain $\mathcal{D}$ is not empty, it is returned to step 1.

The gathered unified partitions cover all confined $\mathrm{h_C}$-vectors so that, together with confinement mapping, an efficient handling of any $\mathrm{h_C}$-vector is possible.

Bibliography

[Absil et al., 2008] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, Princeton, 2008.

[Adair and Suvaci, 2000] J. H. Adair and E. Suvaci. Morphological control of particles. *Current Opinion in Colloid & Interface Science*, 5:160–167, 2000.

[Adragni et al., 2012] K. P. Adragni, R. D. Cook, and S. Wu. GrassmannOptim: An R package for Grassmann manifold optimization. *Journal of Statistical Software*, 50:1–18, 2012.

[Al-Rousan et al., 2007] T. Al-Rousan, E. Masad, E. Tutumluer, and T. Pan. Evaluation of image analysis techniques for quantifying aggregate shape characteristics. *Construction and Building Materials*, 21:978–990, 2007.

[Alexandru, 1969] H. V. Alexandru. A macroscopic model for the habit of crystals grown from solutions. *Journal of Crystal Growth*, 5:115–124, 1969.

[Barber et al., 1996] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa. The quickhull algorithm for convex hulls. *ACM transactions on Mathematical Software*, 22:469–483, 1996.

[Barret, 1980] P. J. Barret. The shape of rock particles, a criticai review. *Sedimentology*, 27:291–303, 1980.

[Beyer et al., 1995] O. Beyer, H. Hackel, V. Pieper, and J. Tiedge. *Wahrscheinlichkeitsrechnung und mathematische Statistik*. Mathematik für Ingenieure und Naturwissenschaftler. B.G. Teubner Verlagsgesellschaft, Leipzig, 1995.

[Blaschke, 1916] W. Blaschke. *Kreis und Kugel*. Verlag von Veit & Comp., Leipzig, 1916.

[Boerrigter, 2003] S. Boerrigter. *Modeling of crystal morphology – Growth simulation on facets in arbitrary orientations*. PhD thesis, Katholieke Universiteit Nijmegen, Netherlands, 2003.

[Bonnesen and Fenchel, 1934] T. Bonnesen and W. Fenchel. *Theorie der konvexen Körper*. Springer-Verlag, Berlin, 1934.

[Borchardt-Ott, 2009] W. Borchardt-Ott. *Kristallographie*. Springer-Verlag, Berlin, 2009.

[Borchert, 2007] C. Borchert. On multidimensional population balance modeling for faceted crystals. Master's thesis, Otto-von-Guericke-Universität Magdeburg, 2007.

[Borchert, 2012] C. Borchert. *Topics in Crystal Shape Dynamics*. PhD thesis, Otto-von-Guericke-Universität Magdeburg, 2012.

[Borchert and Sundmacher, 2011] C. Borchert and K. Sundmacher. Morphology evolution of crystal populations: Modeling and observation analysis. *Chemical Engineering Science*, 2011.

[Borchert and Sundmacher, 2012] C. Borchert and K. Sundmacher. Efficient formulation of crystal shape evolution equations. *Chemical Engineering Science*, 84: 85–99, 2012. submitted October 2011.

[Borchert et al., 2007] C. Borchert, N. Nere, A. Voigt, K. Sundmacher, and D. Ramkrishna. On the prediction of the evolution of crystal shape distributions through multidimensional population balances. In *Proceedings of the 3rd International Conference on Population Balance Modelling*, 2007.

[Borchert et al., 2008] C. Borchert, N. Nere, D. Ramkrishna, A. Voigt, and K. Sundmacher. Evolution of crystal shape distributions and morphology classification. In *17th International Symposium on Industrial Crystallization*, 2008.

[Borchert et al., 2009] C. Borchert, N. Nere, D. Ramkrishna, A. Voigt, and K. Sundmacher. On the prediction of crystal shape distributions in a steady-state continuous crystallizer. *Chemical Engineering Science*, 64:686–696, 2009.

[Borgström, 1925] L. H. Borgström. Die geometrische Bedingung für die Entstehung von Kombinationen. *Zeitschrift für Krystallographie*, 62:1–12, 1925.

[Bramley et al., 1996] A. S. Bramley, M. J. Hounslow, and R. L. Ryall. Aggregation during precipitation from solution: A method for extracting rates from experimental data. *Journal of Colloid and Interface Science*, 183(530): 155–165, 1996.

[Briesen, 2007] H. Briesen. Model-based analysis of the effect of particle and impact geometry on attrition of brittle material. *Powder Technology*, 178:87–98, 2007.

[Briesen, 2008] H. Briesen. Modeling of suspension crystaliation processes with complex particle characterization, 2008. Habilitation treatise, RWTH Aachen.

[Briesen, 2009a] H. Briesen. Two-dimensional population balance modeling for shape dependent crystal attrition. *Chemical Engineering Science*, 64:661–672, 2009a.

[Briesen, 2009b] H. Briesen. Adaptive moving pivot technique for growth dominated population balance equations. In *19th European Symposium on Computer Aided Process Engineering*, pages 895–900, 2009b.

[Bronstein et al., 2008] I. N. Bronstein, K. A. Semendjajew, G. Musiol, and H. Mühlig. *Taschenbuch der Mathematik*. Verlag Harri Deutsch, Frankfurt am Main, 2008.

[Brunn, 1887] H. Brunn. *Über Ovale und Eiflächen*. PhD thesis, München, 1887.

[Burger and Burge, 2009] W. Burger and M. J. Burge. *Principles of Digital Image Processing*. Springer, London, 2009.

[Cardew, 1985] P. T. Cardew. The growth shape of crystals. *Journal of Crystal Growth*, 73:385–391, 1985.

[Chakraborty and Kumar, 2007] J. Chakraborty and S. Kumar. A new framework for solution of multidimensional population balance equations. *Chemical Engineering Science*, 62:4112–4125, 2007.

[Chakraborty et al., 2010] J. Chakraborty, M. R. Singh, D. Ramkrishna, C. Borchert, and K. Sundmacher. Modeling of crystal morphology distributions. towards crystals with preferred asymmetry. *Chemical Engineering Science*, 65:5676–5686, 2010.

[Chen et al., 1997] S. Chen, B. Merriman, S. Osher, and P. Smereka. A simple level set method for solving Stefan problems. *Journal of Computational Physics*, 135:8–29, 1997.

[Chernov, 1963] A. A. Chernov. The kinetics of the growth forms of crystals. *Soviet Physics – Chrystallography*, 7: 895–898, 1963.

[Choscz, 2012] C. Choscz. Formabhängige modellierung des kristallwachstums von α-laktose monohydrat auf basis eines momentenmodells. Master's thesis, TU München, 2012.

[Christoffersen, 1980] J. Christoffersen. Kinetics of dissolution of calcium hydroxypatite. *Journal of Crystal Growth*, 49:29–44, 1980.

[Christoffersen and Christoffersen, 1976] J. Christoffersen and M. R. Christoffersen. The kinetics of dissolution of calcium sulphate dihydrate in water. *Journal of Crystal Growth*, 35:79–99, 1976.

[Costa et al., 2007] C. B. B. Costa, M. R. W. Maciel, and R. M. Filho. Considerations on the crystallization modeling: Population balance solution. *Computers and Chemical Engineering*, 31:206–218, 2007.

[Desktop Computer, 2009] Standard Desktop Computer. CPU: Intel® Core™2 Duo E8400 3.0 GHz, RAM: 4 GB, OS: Ubuntu 12.04, MATLAB®: R2012A with enforced single thread mode, 2009. 2009 is the year of purchase, the year of computations is 2014.

[Devroye, 1986] Luc Devroye. *Non-Uniform Random Variate Generation*. Springer, New York, 1986.

[Dincer, 2000] T. D. Dincer. *Mechanisms of Lactose Crystallisation*. PhD thesis, Curtin Universtiy of Technology, 2000.

[Edelman et al., 1998] A. Edelman, T. A. Arias, and S. T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, 20:303–353, 1998.

[Edelsbrunner, 1987] H. Edelsbrunner. *Algorithms in Combinatorial Geometry*. Springer, Berlin, 1987.

[Erdemir et al., 2009] D. Erdemir, A. Y. Lee, and A. S. Myerson. Nucleation of crystals from solution: Classical and two-step models. *Accounts of chemical research*, 42:621–629, 2009.

[Fletcher and Powell, 1963] R. Fletcher and M. J. D. Powell. A rapidly convergent descent method for minimization. *The Computer Journal*, 6:163–168, 1963.

[Fonseca, 1991] I. Fonseca. The wulff theorem revisited. *Proceedings of the Royal Society of London Series A-mathematical Physical and Engineering Sciences*, 432: 125–145, 1991.

[Fries et al., 1971] D. C. Fries, S. T. Rao, and M. Sundaralingam. Structural chemistry of carbohydrates. III. crystal and molecular structure of 4-o-β-D-galactopyranosyl-α-D-glucopyranose monohydrate (α-lactose monohydrate). *Acta Crystallographica Section B*, 27:994–1005, 1971.

[Fukuda, 1996] K. Fukuda. Double description method revisited. In *Combinatorics and Computer Science: 8th Franco-Japanese and 4th Franco-Chinese Conference*, 1996.

[Fukuda, 2008] Komei Fukuda. cddlib, 2008. URL `http://www.ifor.math.ethz.ch/~fukuda/cdd_home/`. Version 94f.

[Gadewar and Doherty, 2004] S. B. Gadewar and M. F. Doherty. A dynamic model for evolution of crystal shape. *Journal of Crystal Growth*, 267:239–250, 2004. ISSN 00220248.

[Gahn and Mersmann, 1997] C. Gahn and A. Mersmann. Theoretical prediction and experimental determination of attrition rates. *Chemical Engineering Research and Design*, 75:125–131, 1997.

[Genceli et al., 2007] F. E. Genceli, M. Lutz, A. L. Spek, and G.-J. Witkamp. Crystallization and characterization of a new magnesium sulfate hydrate MgSO4·11h(2)o. *Crystal Growth & Design*, 7:2460–2466, 2007. ISSN 1528-7483.

[Goldfarb, 1970] D. Goldfarb. A family of variable-metric methods derived by variational means. *Mathematics of Computation*, 24:23–26, 1970.

[Gritzmann and Klee, 1993] P. Gritzmann and V. Klee. On the complexity of some basic problems in computational convexity: II. Volume and mixed volumes. Schwerpunktprogramm der Deutschen Forschungsgemeinschaft: Anwendungsbezogene Optimierung und Steuerung 493, Universität Trier, 1993.

[Gruber, 2004] P. M. Gruber. Beziehungen der Konvexgeometrie zu anderen Gebieten, 2004.

[Grünbaum, 2003] B. Grünbaum. *Convex Polytopes*. Springer, Berlin, 2003.

[Gunawan et al., 2004] R. Gunawan, I. Fusman, and R. D. Braatz. High resolution algorithms for multidimensional population balance equations. *AiChE Journal*, 50: 2738–2749, 2004.

[Haario et al., 2001] H. Haario, E. Saksman, and J. Tamminen. An adaptive metropolis algorithm. *Bernoulli*, 7: 223–242, 2001.

[Hadwiger, 1957] H. Hadwiger. *Vorlesungen über Inhalt, Oberfläche und Isoperimetrie*. Springer, Berlin, 1957.

[Hahn, 2005] T. Hahn. CUBA – a library for multidimensional numerical integration. *Computer Physics Communications*, 168:78–95, 2005.

[Hamm, 2008] J. Hamm. *Subspace-based learning with Grassmann kernels*. PhD thesis, University of Pennsylvania, 2008.

[Heine and Pratsinis, 2007] M.C. Heine and S.E. Pratsinis. Brownian coagulation at high concentration. *Langmuir*, 23:9882–9890, 2007.

[Hentschel and Page, 2003] M. L. Hentschel and N. W. Page. Selection of descriptors for particle shape characterization. *Particle & Particle Systems Characterization*, 20:25–38, 2003.

[Herring, 1951] C. Herring. Some theorems on the free energies of crystal surfaces. *Physical Review*, 82:87–93, 1951.

[Hill, 2004] P. J. Hill. Statistics of multiple particle breakage accounting for particle shape. *AiChE Journal*, 50: 937–952, 2004.

[Hough, 1962] P.V.C Hough. Methods and means for recognizing complex patterns, 1962.

[Hounslow et al., 2005] M. J. Hounslow, A. E. Lewis, S. J. Sanders, and R. Bondy. Generic crystallizer model: 1. a model framework for a well-mixed compartment. 51(11): 2942–2955, Nov 2005.

[Hufnagel, 1995] A. Hufnagel. *Algorithmic Problems in Brunn-Minkowski Theory*. PhD thesis, Universität Trier, 1995.

[Illingworth and Kittler, 1988] J. Illingworth and J. Kittler. A survey of the hough transform. *Computer Vision, Graphics and Image Processing*, 44:87–116, 1988.

[Johnsen, 1900] A. Johnsen. *Wachstum und Auflösung der Kristalle*. Engelmann, Leipzig, 1900.

[Jolliffe, 2002] I. T. Jolliffe. *Principal Component Analysis*. Springer, New-York, 2002.

[Karl et al., 1995] W. C. Karl, S. R. Kulkarni, G. C. Verghese, and A. S. Willsky. Local tests for consistency of support hyperplane data. *Journal of Mathematical Imaging and Vision*, 6:249–267, 1995.

[Kempkes et al., 2010] M. Kempkes, T. Vetter, and M. Mazzotti. Measurement of 3d particle size distributions by stereoscopic imaging. *Chemical Engineering Science*, 65:1362–1373, 2010.

[Kiefer, 1953] J. Kiefer. Sequential minimax search for a maximum. *Proceedings of the American Mathematical Society*, 4:502–506, 1953.

[Kjeldsen, 2008] T. H. Kjeldsen. From measuring tool to geometrical object: Minkowski's development of the concept of convex bodies. *Archive for History of Exact Sciences*, 62:59–89, 2008.

[Kovačević et al., 2014] T. Kovačević, A. Reinhold, and H. Briesen. Identifying faceted crystal shape from three-dimensional tomography data. *Journal of Crystal Growth and Design*, 2014. doi: 10.1021/cg401780p.

[Kroese et al., 2011] D. P. Kroese, T. Taimre, and Z. I. Botev. *Handbook of Monte Carlo Methods*. John Wiley & Sons, Hoboken, 2011.

[Kumar and Ramkrishna, 1996] S. Kumar and D. Ramkrishna. On the solution of population balance equations by discretization – I. A fixed pivot technique. *Chemical Engineering Science*, 51:1311–1332, 1996.

[Kumar and Ramkrishna, 1997] S. Kumar and D. Ramkrishna. On the solution of population balance equations by discretization – III. Nucleation, growth and aggregation of particles. *Chemical Engineering Science*, 52:4659–4679, 1997.

[Kvasnica et al., 2004] M. Kvasnica, P. Grieder, and M. Baotić. Multi-parametric toolbox (mpt), 2004. URL `http://control.ee.ethz.ch/~mpt/`.

[Lacmann et al., 1999] R. Lacmann, A. Herden, and C. Mayer. Kinetics of nucleation and crystal growth. *Chemical Engineering Science*, 22:279–289, 1999.

[Lee et al., 2007] J. R. J. Lee, M. L. Smith, and L. N. Smith. A new approach to the three-dimensional quantification of angularity using image analysis of the size and form of coarse aggregates. *Engineering Geology*, 91: 254–264, 2007.

[Leveque, 2002] R. J. Leveque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, Cambridge, 2002.

[Lorenz, 2003] F. Lorenz. *Lineare Algebra I*. Spektrum Akademischer Verlag, Heidelberg, 2003.

[Lorenz, 2005] F. Lorenz. *Lineare Algebra II*. Spektrum Akademischer Verlag, Heidelberg, 2005.

[Ma and Wang, 2008] C. Y. Ma and X. Z. Wang. Crystal growth rate dispersion modeling using morphological population balance. *AiChE Journal*, 54:2321–2334, 2008.

[Ma et al., 2008] C. Y. Ma, X. Z. Wang, and K. J. Roberts. Morphological population balance for modeling crystal growth in face directions. *AiChE Journal*, 54:209–222, 2008.

[Ma et al., 2002] D. L. Ma, Danesh K. Tafti, and R. D. Braatz. High-resolution simulation of multidimensional crystal growth. *Industrial & Engineering Chemistry Research*, 41:6217–6223, 2002.

[Maerz, 2004] N. H. Maerz. Technical and computational aspects of the measurement of aggregate shape by digital image analysis. *Journal of Computing in Civil Engineering*, 18:10–18, 2004.

[Masad, 2000] E. Masad. Unified imaging approach for measuring aggregate angularity and texture. *Computer-Aided Civil and Infrastructure Engineering*, 15:273–280, 2000.

[MATLAB®, 2012] MATLAB®. Matlab® R2012a. The MathWorks, Inc, 2012. URL `http://www.mathworks.com/`.

[Meadhra et al., 1996] R. O. Meadhra, H. J. M. Kramer, and G. M. van Rosmalen. Model for secondary nucleation in a suspension crystallizer. *AICHE Journal*, 42:973–982, 1996.

[Mersmann, 1994] A. Mersmann, editor. *Crystallization Technology Handbook*. Marcel Dekker, New York, 1994.

[Mesbah et al., 2009] A. Mesbah, H. J. M. Kramer, A. E. M. Huesman, and P. M. J. van den Hof. A control oriented study on the numerical solution of the population balance equation for crystallization processes. *Chemical Engineering Science*, 64:4262–4277, 2009.

[Metropolis et al., 1953] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equations of state calculations by fast computing machines. *Journal of Chemical Physics*, 21:1087–1091, 1953.

[Meyer, 1974] W. Meyer. Indecomposable polytopes. *Transactions of the American Mathematical Society*, 190: 77–86, 1974.

[Milnor and Stasheff, 1974] J. W. Milnor and J. D. Stasheff. *Characteristic classes*. Princeton University Press, Princeton, 1974.

[Minkowski, 1911] H. Minkowski. *Gesammelte Abhandlungen von Hermann Minkowski*. AMS Chelsea Publishing, Providence, 1911.

[Motzkin et al., 1953] T. S. Motzkin, H. Raiffa, G. L. Thompson, and R. M. Thrall. *Contributions to theory of games*, volume 2, chapter The double description method. Princeton University Press, Princeton, 1953.

[Mount and Silverman, 1989] D. M. Mount and R. Silverman. Combinatorial and computational aspects of Minkowski decompositions. In *Vision Geometry: proceedings of the AMS special session held October 20–21, 1989, at Hoboken, New Jersey*. American Mathematics Society, 1989.

[Mullin, 2001] J. W. Mullin. *Crystallization*. Butterworth–Heinemann, Oxford, 4 edition, 2001.

[Neubeck and van Gool, 2006] A. Neubeck and L. van Gool. Efficient non-maximum suppression. In *The 18th International Conference of Pattern Recognition*, 2006.

[Nollet et al., 2006] S. Nollet, C. Hilgers, and J. L. Urai. Experimental study of polycrystal growth from an advecting supersaturated fluid in a model fracture. *Geofluids*, 6:185–200, 2006.

[Nyburg et al., 2000] S. Nyburg, J. Steed, S. Aleksovska, and V. Petrusevski. Structure of the alums. I. on the sulfate group disorder in the alpha-alums. *Acta Crystallographica Section B*, 56:204–209, 2000.

[Nývlt and Matuchová, 1976a] J. Nývlt and M. Matuchová. Determination of linear growth rates of crystals (I). Calculation of linear growth rates of individual crystal faces from overall rates. *Kristall und Technik*, 11:149–161, 1976a.

[Nývlt and Matuchová, 1976b] J. Nývlt and M. Matuchová. Determination of linear growth rates of crystals (II). The shape factors method. *Kristall und Technik*, 11: 245–253, 1976b.

[Ohser and Mücklich, 2000] J. Ohser and F. Mücklich. *Statistical Analysis of Microstructures in Materials Science*. John Wiley & Sons, New York, 2000.

[O'Rourke, 1994] J. O'Rourke. *Computational Geometry in C*. Cambridge University Press, Cambridge, 1994.

[Pons et al., 1999] M. N. Pons, H. Vivier, K. Belaroui, B. Bernard-Michel, F. Cordier, D. Oulhanaa, and J.A. Dodds. Particle morphology: from visualisation to measurement. *Powder Technology*, 103:44–57, 1999.

[Powers, 1953] M. C. Powers. A new roundness scale for sedimentary particles. *Journal of Sedimentary Research*, 23:117–119, 1953.

[Preparata and Shamos, 1985] F.P. Preparata and M. I. Shamos. *Computational Geometry. An Introduction*. Springer, New York, 1985.

[Press et al., 2007] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, editors. *Numerical Recipies*. Cambridge University Press, Cambridge, 2007.

[Prince and Willsky, 1990] J. L. Prince and A. S. Willsky. Reconstructing convex sets from support line measurements. *IEEE Journal of Pattern Analysis and Machine Intelligence*, 12:377–389, 1990.

[Prince and Willsky, 1991] J. L. Prince and A. S. Willsky. Convex set reconstruction using prior shape information. *CVGIP: Graphical Models and Image Processing*, 53:413–427, 1991.

[Prywer, 1992] J. Prywer. Influence of second and further neighbors on the presence of faces in crystal habit. *Crystal Research and Technology*, 27:187–193, 1992.

[Prywer, 1995] J. Prywer. Three-dimensional model of faces disappearance in crystal habit. *Journal of Crystal Growth*, 155:254–259, 1995.

[Prywer, 1996] J. Prywer. Three-dimensional model of any shape face disappearance in crystal habit. *Journal of Crystal Growth*, 158:568–575, 1996.

[Puel et al., 2003] F. Puel, G. Févotte, and J. P. Klein. Simulation and analysis of industrial crystallization processes through multidimensional population balance equations. Part 1: a resolution algorithm based on the method of classes. *Chemical Engineering Science*, 58: 3715–3727, 2003.

[Ramkrishna, 1981] D. Ramkrishna. Analysis of population balance – iv the precise connection between monte carlo simulation and population balances. *Chemical Engineering Science*, 36:1203–1209, 1981.

[Ramkrishna, 2000] D. Ramkrishna. *Population Balances*. Academic Press, San Diego, 2000.

[Randolph and Larson, 1971] A.D. Randolph and M. A. Larson. *Theory of Particulate Processes*. Academic Press, New York, 1971.

[Reinhold, 2008] A. Reinhold. Multiskalenmodellierung von aggregationsprozessen. Master's thesis, Otto-von-Guericke-Univerität Magdeburg, August 2008.

[Reinhold and Briesen, 2011] A. Reinhold and H. Briesen. Convex geometry for the morphological modeling and characterization of crystal shapes. *Particle & Particle Systems Characterization*, 28:37–56, 2011. Available online: April 2012.

[Reinhold and Briesen, 2012] A. Reinhold and H. Briesen. Numerical behavior of a multiscale aggregation model – coupling population balances and discrete element models. *Chemical Engineering Science*, 70:165–175, 2012.

[Salvalaglio et al., 2013] M. Salvalaglio, T. Vetter, M. Mazzotti, and M. Parrinello. Controlling and predicting crystal shapes: The case of urea. *Angewandte Chemie – International Edition*, 52:13369–13372, 2013.

[Samad et al., 2011] N. A. F. A Samad, R. Singh, G. Sin, K. V. Gernaey, and R. Gani. A generic multi-dimensional model-based system for batch cooling crystallization processes. *Computers and Chemical Engineering*, 35:828–843, 2011.

[Sands, 1993] D. E. Sands. *Introduction to Crystallography*. Dover Publications, New York, 1993.

[Santaló, 1976] L. A. Santaló. *Integral Geometry and Geometric Probability*. Addison-Wesley, London, 1976.

[Schnaitter, 2011] M. Schnaitter. Schätzfehler für minkowski-funktionale von 3-dimensionalen voxelbildern. Master's thesis, Technische Universität München, 2011.

[Schneider, 2008] R. Schneider. *Convex Bodies: The Brunn–Minkowski Theory*. Cambridge University Press, Cambridge, 2008.

[Schorsch et al., 2012] S. Schorsch, T. Vetter, and M. Mazzotti. Measuring multidimensional particle size distributions during crystallization. *Chemical Engineering Science*, 77:130–142, 2012.

[Schorsch et al., 2014] S. Schorsch, D. R. Ochsenbein, T. Vetter, M. Morari, and M. Mazzotti. High accuracy online measurement of multidimensional particle size distributions during crystallization. *Chemical Engineering Science*, 105:155–168, 2014.

[Schürer, 2003] R. Schürer. A comparison between (quasi-)monte carlo and cubature rule based methods for solving high-dimensional integration problems. *Mathematics and Computers in Simulation*, 62:509–517, 2003.

[Serra, 1982] J. Serra. *Image Analysis and Mathematical Morphology*. Academic Press, 1 edition, 1982.

[Shephard, 1963] G. C. Shephard. Decomposable convex polyhedra. *Mathematica*, 10:89–95, 1963.

[Singh, 2013] M. Singh. *Towards the control of crystal shape and morphology distributions in crystallizers*. PhD thesis, Purdue University, 2013.

[Singh and Ramkrishna, 2013] M. R. Singh and D. Ramkrishna. A comprehensive approach to predicting crystal morphology distributions with population balances. *Crystal Growth & Design*, 13:1397–1411, 2013.

[Singh et al., 2012] M. R. Singh, J. Chakraborty, N. Nere, H.-H. Tung, S. Bordawekar, and D. Ramkrishna. Image-analysis-based method for 3d crystal morphology measurement and polymorph identification using confocal microscopy. *Crystal Growth & Design*, 12:3735–3748, 2012.

[Singh et al., 2013] M. R. Singh, P. Verma, H.-H. Tung, S. Bordawekar, and D. Ramkrishna. Screening crystal morphologies from crystal structure. *Crystal Growth & Design*, 13:1390–1396, 2013.

[Smith and Matsoukas, 1998] M. Smith and T. Matsoukas. Constant-number Monte Carlo simulation of population balances. *Chemical Engineering Science*, 53: 1777–1786, 1998.

[Snyder and Doherty, 2007] R. C. Snyder and M. F. Doherty. Faceted crystal shape evolution during dissolution or growth. *AiChE Journal*, 53:1337–1348, 2007.

[Snyder et al., 2007] R. C. Snyder, S. Studener, and M. F. Doherty. Manipulation of crystal shape by cycles of growth and dissolution. *American Institute of Chemical Engineers*, 53:1510–1517, 2007.

[Szoplik, 1996] T. Szoplik, editor. *Morphological Image Processing*, volume MS 127 of *SPIE Milestone Series*. SPIE Optical Engineering Press, Bellingham, 1996.

[Szurgot et al., 1991] M. Szurgot, J., and Prywer. Growth velocities and disappearance of faces of crystals. *Crystal Research and Technology*, 26:147–153, 1991.

[Taylor et al., 1992] J. E. Taylor, J. W. Cahn, and C. A. Handwerker. I–geometric models of crystal growth. *Acta Metallurgica et Materialia*, 40:1443–1474, 1992.

[Trzeciak et al., 2004] T. M. Trzeciak, A. Podgórski, and J. C. M. Marijnissen. Langevin dynamics simulation of aerosol coagulation in highly concentrated systems. *Inzynieria Chemiczna i Procesowa*, 25:1741–1746, 2004.

[Ugray et al., 2007] Z. Ugray, L. Lasdon, F. Glover, J. Kelly, and R. Martí. Scatter search and local nlp solvers: A multistart framework for global optimization. *INFORMS Journal on Computing*, 19:328–340, 2007.

[van Oosterhout and van Rosmalen, 1980] G.W. van Oosterhout and G.M. van Rosmalen. Analysis of kinetic experiments on growth and dissolution of crystals in suspension. *Journal of Crystal Growth*, 48:464–468, 1980.

[Variankaval and Cote, 2008] N. Variankaval and A. S. Cote. From form to function: Crystallization of active pharmaceutical ingredients. *AiChE Journal*, 54:1682–1688, 2008.

[Vermolen et al., 2005] F. J. Vermolen, C. Vuik, E. Javierre, and S. van der Zwaag. Review on some Stefan problems for particle dissolution in solid metallic alloys. *Nonlinear Analysis Modelling and Control*, 10: 257–292, 2005.

[Wadell, 1932] H. Wadell. Volume, shape, and roundness of rock particles. *The Journal of Geology*, 40:443–451, 1932.

[Wang and Ma, 2009] X. Z. Wang and C. Y. Ma. Morphological population balance model in principal component space. *AiChE Journal*, 55:2370–2381, 2009.

[Westhoff and Kramer, 2012] GM Westhoff and HJM Kramer. Scale-up of suspension crystallisers using a predictive model framework. *Chem Eng Sci*, 77:26–34, 2012.

[Wong, 1967] Y.-C. Wong. Differential geometry of Grassmann manifolds. *Proceedings of the National Academy of Sciences of the United States of America*, 57:589–594, 1967.

[Wulff, 1901] G. Wulff. Zur frage der geschwindigkeit des wachstums und der auflösung der krystallflächen. *Zeitschrift für Krystallographie und Mineralogie*, 34:449–530, 1901.

[Zhang et al., 2006] Y. Zhang, J. P. Sizemore, and M. F. Doherty. Shape evolution of 3-dimensional faceted crystals. *AiChE Journal*, 52:1906–1915, 2006.

Index