Efficient Contact Mode Enumeration in 3D

Eric Huang, Xianyi Cheng, and Matthew T. Mason

Carnegie Mellon University, Pittsburgh PA 15213, USA
erichl, xianyic@andrew.cmu.edu; matt.mason@cs.cmu.edu

Abstract. In a hybrid dynamical system with multiple rigid bodies, the relative motions of the contact points on two colliding bodies may be classified as separating, sticking (moving together), or sliding. Given a physical contact model, the active contact modes determine the dynamic equations of motion. Analogously, the set of all possible (valid) contact mode assignments enumerates the set of all possible dynamical flows of the hybrid dynamical system at a given state. Naturally, queries about the kinematics or dynamics of the system can be framed as computations over the set of possible contact modes. This paper investigates efficient ways to compute this set.

To that end, we have developed the first efficient 3D contact mode enumeration algorithm. The algorithm is exponential in the degrees of freedom of the system and polynomial in the number of contacts. The exponential term is unavoidable and an example is provided. Prior work in this area has only demonstrated efficient contact mode enumeration in 2D for a single rigid body. We validated our algorithm on peg-in-hole, boxes against walls, and a robot hand grasping an ellipse. Our experimental results indicate real-time contact mode enumeration is possible for small to medium sized systems. Finally, this paper concludes with a discussion of possible related application areas for future work. Ultimately, the goal of this paper is to provide a novel computational tool for researchers to use to simulate, analyze, and control robotic systems that make and break contact with the environment.

Keywords: Contact Mode Enumeration · Combinatorial Geometry · Dynamics · Kinematics

1 Introduction

When a moving robot contacts its environment, the points of the resulting contact manifold may be sliding, sticking, or separating. Under the Coulomb model of friction, the frictional force is either opposite to the velocity of a sliding contact point, oriented in any direction for a sticking contact point, or zero for a separating contact point. As each individual contact mode imparts complementary dynamic equations, the set of all (valid) contact mode assignments enumerates the set of all possible flows for the system [13]. (Here, by valid we mean kinematically feasible, i.e. there exists a generalized velocity $\dot{q}$ that generates the correct mode at each contact point.) Given the one-to-one mapping between contact modes and dynamics, we advocate that efficient contact mode enumeration will be a useful tool for the simulation, analysis, and control of robotic systems that make and break contact with the environment.

Our main contribution is to report the first efficient algorithm for contact mode enumeration in 3D which is exponential in the number of degrees of freedoms of the system and polynomial in the number of contact points. Letting d and n be those two numbers, respectively, then our algorithm enumerates all feasible contact modes in $O(n^{d^2/2+2.5d})$ time. By efficient, we mean the first algorithm that is polynomial in the number of contact points. The exponent in d is unavoidable (see Section 6 for an example). Our algorithm can enumerate all of the following:

- contact modes that are contacting or separating,
- contact modes that are sticking or sliding in some set of directions,
- contact modes for linear and nonlinear friction cones,
- contact modes involving multiple objects.

In this paper, we address the contact mode enumeration problem as a combinatorial geometry problem, and the key concern is to compute the combinatorial structure of convex hulls and hyperplane arrangements. Leveraging the rigid body kinematics, our method performs faster enumeration by dividing the problem into two steps: contacting/separating enumeration and sliding/sticking enumeration. The results obtained after the two steps are polyhedral convex cones of valid object motions and their corresponding contact modes.

2 Related Work

This section discusses related work specific to contact mode enumeration and our algorithm (for related work with respect to application areas refer to Section 7). Mason [17] sketched an algorithm for contact mode enumeration in 2D for a single rigid body which intersects the positive (negative) rotation centers on the positive (negative) oriented plane and intersects the rotation centers at infinity on the equator. Though Mason [17] upper-bounded the number of modes at $O(n^2)$, by our analysis, the algorithm's runtime is actually $O(n \log n)$ and the correct number of modes is $\Theta(n)$. Unfortunately, the oriented plane technique does not generalize to contact mode enumeration in 3D. Later, Haas-Heger et al [11] independently published an algorithm for partial contact mode enumeration in 2D. There, they interpret the feasible modes as the regions of an arrangement of hyperplanes in 3D. However, Haas-Heger et al [11]'s algorithm is at least $\Omega(n^4)$ and does not enumerate separating modes. The existence of an efficient contact mode enumeration algorithm in 2D does not appear to wide-spread knowledge. For instance, Greenfield et al [10] used the exponential time algorithm for contact mode enumeration in 2D. Disregarding these issues, Haas-Heger et al [11]'s work inspired us to investigate hyperplane arrangements in higher dimensions for our algorithm. To the best of our knowledge, our algorithm is the first method for contact mode enumeration in 3D.

It is well known that frictional contact problems can be modeled as a complementarity problem or equivalently, a variational inequality [8]. Within that theory, it is known that the normal manifold (which is a linear hyperplane arrangement) divides the solution space of an affine variational inequality [8]. Not surprisingly, we found related papers in other fields containing problems that can be modeled as variational inequalities [9, 21]. For example, in the study of digital controllers and power electronics, Geyer et al [9] proposed a mode enumeration algorithm for compositional hybrid systems based on the reverse search technique of Avis and Fukuda [1]. The contribution of our work (and theirs) is in presenting the theory in an understandable manner for our field and optimizing the relevant algorithms for our specific problem formulation.

3 Kinematics of Contact

3.1 Rigit Body Transformation

Let $se(3) = \{(v, \omega) : v \in \mathbb{R}^3, \omega \in \mathbb{R}^3\}$ be the *twist* coordinates of the Euclidean group $SE(3)$. There is a mapping from $\xi \in se(3)$ to $g \in SE(3)$ given by the exponential map $g = \exp \hat{\xi}$, where the hat operator takes

$$\hat{\xi} \in se(3) \to \begin{bmatrix}\hat{\omega} & v\\ 0 & 0\end{bmatrix}, \quad \hat{\omega} \in \mathbb{R}^3 \to \begin{bmatrix}0 & -\omega_3 & \omega_2\\ \omega_3 & 0 & -\omega_1\\ -\omega_2 & \omega_1 & 0\end{bmatrix}. \tag{1}$$

Note, the vee operator takes $\hat{\xi}^\vee \to \xi$, $\hat{\omega}^\vee \to \omega$. Let

$$g_{ab} = \begin{bmatrix}R_{ab} & p_{ab}\\ 0 & 1\end{bmatrix} \in SE(3), \quad \mathrm{Ad}_{g_{ab}} = \begin{bmatrix}R_{ab} & \hat{p}_{ab}R_{ab}\\ 0 & R\end{bmatrix} \in \mathbb{R}^{6\times 6} \tag{2}$$

be a transformation from frame b to frame a and the adjoint of that transformation, respectively. We define the instantaneous *spatial velocity* and *body velocity* of g_{ab} as

$$V^s_{ab} = \left[\dot{g}_{ab} g^{-1}_{ab}\right]^\vee, \quad V^b_{ab} = \left[g^{-1}_{ab}\dot{g}_{ab}\right]^\vee \tag{3}$$

respectively, both of which are in $se(3)$. The adjoint of $Ad_{g_{ab}}$ relates body and spatial velocities linearly $V^s_{ab} = \mathrm{Ad}_{g_{ab}} V^b_{ab}$.

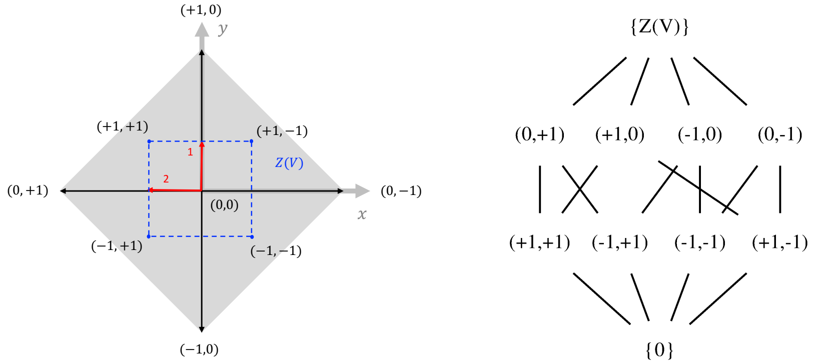
3.2 Contact Normal Velocity

Let $g_{oc} \in SE(3)$ be the transformation from the contact frame to the object frame. By convention, we fix the contact normal $n \in \mathbb{R}^3$, also known as the z-axis of g_{oc}, to point in the direction opposite the object's surface normal at the contact point p_{oc}. Let $g_{wo} \in SE(3)$ and $g_{oc} \in SE(3)$ be the transformations from the object frame to the world frame and the contact frame to the object frame, respectively. Given an object body velocity $V_o \in se(3)$, we can compute the body velocity $V_c \in se(3)$ of the contact frame using

$$V_c = \mathrm{Ad}^{-1}_{g_{oc}} V_o = \begin{bmatrix}R^T_{oc} & -R^T_{oc}\hat{p}_{oc}\\ 0 & R^T_{oc}\end{bmatrix} V_o. \tag{4}$$

Let $\phi \in \mathbb{R}$ be the contact distance as measured along the contact normal, then from (4) the separating velocity is

$$\dot{\phi} = \begin{bmatrix}n & -n\hat{p}_{oc}\end{bmatrix} V_o. \tag{5}$$



(a) $\mathcal{H}(V)$ and $Z(V)$

(b) Face lattice of $Z(V)$

Fig. 1: (a) Hyperplane arrangement $\mathcal{H}(V)$ and its associated zonotope $Z(V)$ (the blue dotted square). Red arrows are the normals of the hyperplanes. Gray shaded areas, black rays and the origin are the faces of $\mathcal{H}(V)$, correspondingly sharing the same sign vectors with the vertices, lines and the interior of $Z(V)$. (b) The face lattice of $Z(V)$, which has the opposite structure with the face lattice of $\mathcal{H}(V)$.

Note that when ϕ is negative, the objects are penetrating. Similarly, Let $t_x \in \mathbb{R}^3$ and $t_y \in \mathbb{R}^3$ be the basis of contact tangent plane, the contact tangential velocity V_c^t is

$$V_c^t = \begin{bmatrix} t_x & -t_x\hat{p}_{oc} \\ t_y & -t_y\hat{p}_{oc} \end{bmatrix} V_o. \tag{6}$$

For a contact between two objects o_1 and o_2, the contact velocity is the relative velocity of the contact on one object to the other. Without loss of generality, we have $V_c = \mathrm{Ad}_{g_{o_1c}}^{-1} V_{o_1} - \mathrm{Ad}_{g_{o_2c}}^{-1} V_{o_2}$.

3.3 Contact Tangent Velocity Approximation

In this paper, sliding modes identifies possible contact tangent velocity directions. We approximate infinite tangent velocity directions by dividing the tangent plane into sectors of equal angles. Let the normals of n_t dividing planes be written as $T = [t_1, \cdots, t_{n_t}]$ where $t_i = \left[\cos\frac{i\pi}{n_t}, \sin\frac{i\pi}{n_t}, 0\right]$, contact tangent velocities can be classified into $4n_t + 1$ contact modes, specified by sign vectors in $\{+1, 0, -1\}^n$. Figure 1(b), which serves as an example of hyperplane arrangement later in Section 4.1, can also be considered as a tangent plane division with $n_t = 2$.

The dividing planes of contact point p are transformed into the object frame as $[T, -T\hat{p}]$ through Equation 6. With n_c contacts, all the dividing planes transformed into the object frame are a set of $n_d \cdot n_c$ hyperplanes that intersect at the origin and divide the space of object velocity in $\mathbb{R}^6$. Moreover, this approximation is essential to applications using linear approximations of the Coulomb friction model [24]. For more fine grained approximations, we can use a larger n_t.

4 Convex and Combinatorial Geometry

This section briefly introduces convex polytopes and hyperplane arrangements. The normal velocity and tangent velocity equations define halfspaces which carve $\mathbb{R}^d$ into regions of valid object motions. Identifying and labeling these regions are at heart combinatorial problems about convex polytopes and hyperplane arrangements, respectively. Note that this section only gives the reader a minimal review of this rich subject. One can refer to Ziegler [27] and Edelsbrunner [7] for a comprehensive introduction.

4.1 Convex Polytopes

Let $P \subseteq \mathbb{R}^d$ be a convex set. This work primarily uses the following two classes of convex sets. The $\mathcal{H}\text{-}polyhedron$ is an intersection of closed halfspaces $\mathcal{H}(A, z) = \{x \in \mathbb{R}^d : Ax \leq z\}$. The $\mathcal{V}\text{-}polytope$ is the convex hull of a finite point set $\mathcal{V}(A) = \{x \in \mathbb{R}^d : x = At, t \geq 0, \mathbf{1}t = 1\}$. Let $P \subseteq \mathbb{R}^d$ be a convex polyhedron. Let a *face* of P be any set of the form

$$F = \{x \in P : cx = c_0\} \tag{7}$$

where $cx \leq c_0$ is true for all $x \in P$. Suppose further that P is an $\mathcal{H}$-polytope. Then the sign of face F with respect to a halfspace (a, z) is

$$\text{sgn}_a(F) = \begin{cases} +1 & \text{if} \quad a \cdot x < z \\ 0 & \text{if} \quad a \cdot x = z \end{cases} \tag{8}$$

for any $x \in F$. Naturally, the sign vector of a face is given by

$$\text{sgn}_A(F) = [\text{sgn}_{a_1}(F), \ldots, \text{sgn}_{a_n}(F)]. \tag{9}$$

4.2 Hyperplane Arrangements

A hyperplane through the origin, i.e. a *linear hyperplane*, is the set $H = \{x \in \mathbb{R}^d : h \cdot x = 0\}$ parameterized by the normal vector $h \in \mathbb{R}^d$. In this work, we will refer to a linear hyperplane by its normal vector h. Given a point $p \in \mathbb{R}^d$, let the *sign* of the point relative to a hyperplane h be

$$\text{sgn}_h(p) = \begin{cases} +1 & \text{if} \quad h \cdot p > 0 \\ 0 & \text{if} \quad h \cdot p = 0 \\ -1 & \text{if} \quad h \cdot p < 0 \end{cases} \tag{10}$$

Let a *hyperplane arrangement* be defined as a set of linear hyperplanes $\mathcal{A} = [h_1, \ldots, h_k]$ which dissect $\mathcal{R}^d$ into connected, polyhedral convex cones of different dimensions. Let the sign of a point p with respect to the hyperplane arrangement $\mathcal{A}$ be defined as

$$\text{sgn}_{\mathcal{A}}(p) = \left[\text{sgn}_{h_1}(p), \ldots, \text{sgn}_{h_k}(p)\right]. \tag{11}$$

The sign vectors of the points in $\mathbb{R}^d$ define equivalence classes known as the *faces* of $\mathcal{A}$. That is, given a sign vector $s \in \{+1, 0, -1\}^k$, the associated face is

$$F = \{p \in \mathbb{R}^d : \text{sgn}_{\mathcal{A}}(p) = s\}. \tag{12}$$

A simple example of a 2D linear hyperplane arrangement is given in Figure 1a, and the faces are annotated with their corresponding sign vectors: regions have sign vectors $(+1,+1)$, $(+1,-1)$, $(-1,+1)$, $(-1,-1)$; rays have sign vectors $(+1,0)$, $(0,+1)$, $(-1,0)$, $(0,-1)$; the origin has the sign vector $(0,0)$.

4.3 Face Lattice

We can define a partial ordering over the sign vectors of polytopes or hyperplane arrangements. The *face lattice* $L(P)$ of a polytope P (or arrangement $\mathcal{A}$) is a *partially ordered set* over the sign vectors of the faces of P. Define the partial order over the set of signs $\{+1,0,-1\}$ to be

$$0<+1, \quad 0<-1, \quad +1 \parallel -1 \text{ (incomparable)}. \tag{13}$$

Let u and v be two sign vectors. Then $u \leq v$ if and only if $u_i \leq v_i$ for all indices. By convention, every face lattice is equipped with an unique maximal element $\{P\}$ (sometimes written $\{1\}$) and minimal element $\{\varnothing\}$. Figure 1b visualizes the face lattice of the blue square in Figure 1a.

The face lattice $L(P)$ contains $d+1$ proper *ranks*. For $k \in [0,\ldots,d]$, the k-th rank contains the faces of dimension k. The dimension of a face is defined as the dimension of its *affine hull* $\dim(F)=\dim(\operatorname{aff}(F))$, where $\operatorname{aff}(F)=\{\sum \alpha_i x_i : x_i \in S, \sum \alpha_i = 1\}$. The faces of dimensions 0, 1, $\dim(P)-2$, and $\dim(P)-1$ are called *vertices*, *edges*, *ridges*, and *facets*, respectively.

4.4 Polarity

The polar polytope $P^{\triangle}$ of a polytope $P \subseteq \mathbb{R}^d$ is defined by

$$P^{\triangle}=\{c \in \mathbb{R}^d : c^T x \leq 1, \forall x \in P\} \subseteq \mathbb{R}^d \tag{14}$$

In this definition, we assume that **0** is in the interior of the polytope P without loss of generality. Therefore, the polar of a $\mathcal{V}$-polytope $\mathcal{V}(A)$ with $0 \in \operatorname{int} P$ is the $\mathcal{H}$-polytope $H(A,1)$, and vice versa. Polar polytopes are useful because P and P^{Δ} share the same combinatorial structure. Specifically, the face lattice of the polar polytope $P^{\triangle}$ is the opposite of the face lattice of P:

$$L(P^{\triangle})=L(P)^{op} \tag{15}$$

and there is a bijection between the faces

$$\begin{aligned} \emptyset &\longleftrightarrow P \\ vertices &\longleftrightarrow facets \\ edges &\longleftrightarrow ridges \\ \ldots &\longleftrightarrow \ldots \end{aligned} \tag{16}$$

4.5 Zonotopes

Zonotopes are a special type of polytope which are combinatorially equivalent to linear hyperplane arrangements. Recall that the *Minkowski sum* of sets X and Y is given by $X \oplus Y = \{x + y : x \in X, y \in Y\}$. We can define a zonotope as the Minkowski sum of a set of line segments

$$Z(V) = [-v_1, v_1] \oplus \cdots \oplus [-v_k, v_k], \tag{17}$$

where $V = [v_1, \ldots, v_k] \in \mathbb{R}^{d \times k}$. We can map each face F of a zonotope to an unique sign vector. Let $p = \sum \lambda_i v_i \in \operatorname{int} F$ (any interior point will do). Then the sign with respect to v_i is

$$\operatorname{sgn}_{v_i}(F) = \begin{cases} +1 & \text{if} \quad \lambda_i = +1 \\ 0 & \text{if} \quad -1 < \lambda_i < 1 \\ -1 & \text{if} \quad \lambda_i = -1, \end{cases} \tag{18}$$

and the sign vector is $\operatorname{sgn}_Z(F) = \left[\operatorname{sgn}_{v_1}(F), \ldots, \operatorname{sgn}_{v_k}(F)\right]$. From Corollary 7.17 in Ziegler [27], there is a bijection between the sign vectors of $\mathcal{A}(V)$ and $Z(V)$. We have the identification of face lattices

$$L(Z(V)) \longleftrightarrow L(Z(V)^{\triangle}) \longleftrightarrow L(\mathcal{A}(V)). \tag{19}$$

For example, there is a correspondence between the facets of $Z(V)$, the vertices of $Z(V)^{\triangle}$, and the rays (unbounded edges) of $\mathcal{A}(V)$. Figure 1a shows a 2D hyperplane arrangement and its associated zonotope.

5 Contact Mode Enumeration in 3D

This section describes our algorithm for contact mode enumeration in 3D. The algorithm is presented in three parts. The first subsection covers contacting/separating mode enumeration. The second subsection covers sliding/sticking mode enumeration given an assignment of contacting/separating modes. The algorithm in this subsection constructs the hyperplane arrangement by incrementally building the associated polar zonotope using the Minkowski sum. The third subsection covers full mode enumeration by combining the previously stated algorithms.

5.1 Contacting/Separating Mode Enumeration

The contacting/separating mode enumeration algorithm, or CS-Enumerate, takes as input the normal velocity constraint equations $A \in \mathbb{R}^{n \times d}$ (see Section 3) and generates a list of valid contacting/separating sign vectors of the form $m \in \{0, +1\}^n$. The algorithm presented in this subsection is based on taking the convex hull in polar form of the polytope associated with the normal velocity constraints. The pseudo-code is listed in Algorithm 1 and we provide explanations for each of the steps below.

Find an interior point: The polar form P^{Δ} of a polyhedron P is defined only when $0 \in \operatorname{relint}(P)$. However, 0 is on the boundary of the polyhedral cone $\mathcal{H}(A, 0)$

Algorithm 1 C/S Mode Enumeration

Input: Contact points: $p_1, \dots, p_k$; Contact normals: $n_1, \dots, n_k$
Output: Contacting/Separating Modes: M_{cs}

```
function CS-ENUMERATE($P$, $N$)
    $A \leftarrow$ ADD-HYPERPLANES($([n_i, -n_i\hat{p_i}])$) for all contacts $i \in [1, \dots, k]$
    $r \leftarrow$ INTERIOR-POINT($(\mathcal{H}(A, 0))$)
    $A$, $r \leftarrow$ PROJECT-TO-NULLSPACE($(A, r)$)
    $\mathcal{V}(A^T) \leftarrow$ POLAR($(\mathcal{H}(A, 0), r)$)
    $A^T \leftarrow$ PROJECT-TO-AFFINE-SUBSPACE($(A^T)$)
    $M \leftarrow$ CONV-HULL($(\mathcal{V}(A^T))$)
    $L \leftarrow$ FACE-LATTICE($(M)$)
    $M_{cs}$, $m \leftarrow \varnothing$, []
    for $k \in \{0, \dots, d-1\}$ do                ▷ The face lattice has $d$ proper ranks.
        for $f \in L[k]$ do     ▷ Each face in $L[k]$ as defined by its vertex (= constraint) set.
            $m[f] \leftarrow 0$           ▷ A vertex $v \in f \Rightarrow$ sign of normal vel at that contact is 0
            $m[A^T \setminus f] \leftarrow 1$    ▷ A vertex $v \notin f \Rightarrow$ sign of normal vel at that contact is $+1$
            $M_{cs} \leftarrow M_{cs} \cup \{m\}$
    return $M_{cs}$
```

defined by our normal velocity constraints. Therefore, our first step is to find a point $r \in \operatorname{relint}(\mathcal{H}(A, 0))$. This is a classical problem in linear programming, and for our implementation, it amounts to solving the following linear program

$$\min_{r,c} \quad \begin{bmatrix} 0 & -1 \end{bmatrix} \cdot \begin{bmatrix} r \\ c \end{bmatrix} \tag{20}$$

$$\text{s.t.} \quad Ar + c \leq 0, \; c \geq 0 \tag{21}$$

$$\|r\|_\infty \leq 1, \tag{22}$$

where $\|r\|_\infty \leq 1$ constrains r to be within the hypercube in $\mathbb{R}^d$. Note that if the solution to the linear program is $r = 0$, then the only valid mode is all-contacting and the algorithm can terminate early. The above method was adapted from [22] to handle cones.

Project to contact planes: If the interior point is on the boundary for a subset of the normal velocity constraints, then that subset of contact points must always be in contact (for example, a box sandwiched between two walls). Let A_c be the contacting normal velocity constraints and $A_o = A/A_c$. Then we map A_o into the nullspace of A_c, like so $A_o = A_o \cdot \textsc{Null}(A_c)$, to reduce the dimension of the problem. We also express the interior point as coordinates in the null space.

Convert to polar form: Given a strictly interior point r, we translate the origin to r, resulting in the new $\mathcal{H}$-polyhedron $P = \mathcal{H}(A, -Ar)$. Next we normalize the inequalities so that $P = \mathcal{H}(A, 1)$ and obtain the polar polytope $P^\Delta = \mathcal{V}(A^T)$.

Project to affine subspace: The affine dimension of the polar polytope $\dim \operatorname{aff} P^\Delta$ may not necessarily be equal to the dimension of the ambient space $\mathbb{R}^d$. In this situation, we project P^Δ into its affine subspace $\operatorname{aff} P^\Delta = \{A^T v : 1^T v = 1\}$ and further reduce the dimensionality of convex hull. Recall that an affine space can also be expressed as a linear space plus a translate, i.e. $\operatorname{aff} P^\Delta = \{Vx + z\}$ for some V and z. If $0 \notin \operatorname{aff} P^\Delta$, then $z \neq 0$ and we translate the affine space until it contains the origin. Now that $0 \in \operatorname{aff} P^{\Delta}$, $\operatorname{aff} P^{\Delta}$ is a linear subspace and we project each point (column vector) in A^T to coordinates on the column space of A^T.

Get facet-vertex incidence matrix: Next, the algorithm constructs the vertex-facet incidence matrix M of $P^{\Delta} = \mathcal{V}(A^T)$ by using a convex-hull algorithm. The vertex-facet incidence matrix is a matrix $M \in \{0,1\}^{n_v \times n_f}$, where n_v and n_f are the number of vertices and facets, respectively. We associate the vertices and facets with the index sets $I_V = \{1, \ldots, n_v\}$ and $I_F = \{1, \ldots, n_f\}$, so that $m_{vf} = 1$ if facet f contains v and $m_{fv} = 0$ otherwise. The vertex-facet incidence matrix is a standard return value from convex hull algorithms such as `qhull` [3].

Build face lattice: Given the facet-vertex incidence matrix M of P^{Δ}, we can construct the face lattice $L(P^{\Delta})$ using the algorithm of Kaibel and Pfetsch [14]. Their method is based on find the closed sets (= faces) with respect to a closure map defined over vertex sets. Obviously, each face is uniquely represented by its vertex set.

Convert faces to mode strings: Finally, we construct contacting/separating mode strings using the vertex sets associated with each face f in $L(P^{\Delta})$. By polarity, each vertex in f corresponds to the hyperplane $\{x : ax = 0\}$ defined by a normal velocity constraint. Therefore, we can read off the mode string by assigning contacting modes to every vertex in f and separating modes to every vertex not in f.

Theorem 1. *For a set of n contacts in a system of colliding bodies with d degrees of freedom, Algorithm 1 enumerates the possible contacting and separating modes in $O(d \cdot n^{d+1} + l(n,d))$ time.*

Proof. We analyze correctness first before complexity. The proof is simple and relies on the combinatorial equivalences between

$$\text{CS-MODES} \leftrightarrow L(\mathcal{H}(A,0)) \leftrightarrow L(\mathcal{H}(A_{\text{int}},1)) \leftrightarrow L(\mathcal{V}(A_{\text{int}}^T)). \tag{23}$$

First, we show that $\mathcal{H}(A,0)$ and $\mathcal{H}(A_{\text{int}},1)$ are affinely isomorphic and thus, combinatorially equivalent [27]. Two polytopes P and Q are *affinely isomorphic* if there exists an affine map $f : \mathbb{R}^d \to \mathbb{R}^e$ that is a bijection between the vertices of the two polytopes. By inspection, the operations $P \cap \operatorname{aff}(P)$ and $P + r$ preserve the extremal points (vertices). Finally, re-scaling the inequalities does not affect the underlying polytope.

For this next paragraph, let us define $P = \mathcal{H}(A_{\text{int}},1)$ and $P^{\Delta} = \mathcal{V}(A_{\text{int}}^T)$. Our aim is to show the first and third bijections in (23). Let $F \in L(P^{\Delta})$ be identified by its vertex set $V(F) = \{a : a \cap F \neq \varnothing, a \in \operatorname{vert}(P^{\Delta})\}$ and recall that $\operatorname{vert}(P^{\Delta}) \subseteq \operatorname{col}(A_{\text{int}}^T) = \operatorname{row}(A_{\text{int}})$. (That is, each vertex of P^{Δ} corresponds to a facet of P, i.e. a normal velocity constraint.) Then by Corollary 2.13 of Ziegler [27], there is a bijection $L(P^{\Delta}) \leftrightarrow L(P)$ from F to $F^{\diamond}$ such that

$$F^{\diamond} = \{x : A_{\text{int}}x \leq 1, ax = 1, \forall a \in V(F)\} \tag{24}$$

is a non-empty face of P. Because face lattice of a polytope is coatomic, we can uniquely specify its proper elements as meets (intersections) $a_1 \wedge \ldots \wedge a_k$ of its coatoms (facets). Therefore, for each $F \in L(P^{\Delta})$, the vertex set $V(F)$ maps bijectively to a valid contacting/separating mode string, and $L(P^{\Delta})$ enumerates the set of all valid contacting/separating modes.

Algorithm 2 S/S Mode Enumeration

Input: Contact points: $p_1, \ldots, p_k$; Contact normals: $n_1, \ldots, n_k$; Contact tangent dividing planes: $T_1, \ldots, T_k$; Contact/Separating Mode: m_{cs}

Output: Sliding Modes: M_{ss}

```
function SS-ENUMERATE($P$, $N$, $T$, $m$)
    $H \leftarrow \varnothing$
    $H \leftarrow$ ADD-HYPERPLANES($[n_i, -n_i\hat{p_i}]$) for all separating contacts $m_{cs}^{(i)} = 1$
    $H \leftarrow$ ADD-HYPERPLANES($[T_j, -T_j\hat{p_j}]$) for all contacting contacts $m_{cs}^{(j)} = 0$
    $H \leftarrow$ PROJECT-TO-NULLSPACE($[n_j, -n_j\hat{p_j}]$) for all contacting contacts $m_{cs}^{(j)} = 0$
    $V, S \leftarrow$ GET-ZONOTOPE-VERTICES($H$)
    $L \leftarrow$ FACE-LATTICE($\mathcal{V}(V)$)
    $\mathcal{F} \leftarrow F \in L$ with positive signs $\{+1\}$ for all separating contact hyperplanes
    $M_{ss} \leftarrow$ GET-SIGN-VECTOR($\mathcal{F}, S$)
    return $M_{ss}$
function GET-ZONOTOPE-VERTICES(H)
    $V,\ S \leftarrow$ [0], $\varnothing$
    for $h \in H$ do
        $V',\ S' \leftarrow [\,],\ \varnothing$
        for $v \in V$ do
            $V' \leftarrow$ ADD-POINTS($v + h, v - h$)
            $S' \leftarrow$ ADD-SIGN-VECTORS($(S(v), +1), (S(v), -1)$)
        $V \leftarrow$ CONVEX-HULL($V'$)
        $S \leftarrow S'(V)$
    return V, S
```

The normal velocity constraint matrix A can be constructed in $O(n \cdot d)$ time. The orthonormal basis and null space can be computed in $O(\min\{n \cdot d^2, n^2 \cdot d\})$ using SVD. An interior point can be computed in time $O(l(n,d))$, where $l(n,d)$ is the cost of linear programming. For a balanced problem like this one (every input point is extremal), quick hull runs in $O(f_{d-1}) = O(n^{d/2})$. The number of k-faces in $L(P)$ is bound by $O(n^{d/2})$. The combinatorial face enumeration algorithm runs in time $O(n \cdot \sum_{k=0}^{d} f_k^2) = O(d \cdot n^{d+1})$. Therefore, the total runtime is $O(d \cdot n^{d+1} + l(n,d))$.

5.2 Sticking/Sliding Mode Enumeration

This section describes our sliding/sticking mode enumeration algorithm, which is listed in Algorithm 2. The inputs to this algorithm are the normal velocity constraints, tangent velocity hyperplanes (see Section 3.3) and the specified contacting/separating mode. The algorithm, SS-ENUMERATE, generates a list of sliding/sticking sign vectors of the form $m_{ss} \in \{-1, 0, +1\}^{n_t}$, where n_t is the total number of tangent velocity hyperplanes. As before we provide explanations for each of the steps below.

Partition the hyperplanes: The goal of our algorithm is to enumerate sliding/ sticking modes for the "contacting" contacts. Given a contacting/separating mode m_{cs}, the normal velocity constraints on the object velocity x are

$$H_c x = 0, \quad H_{sep} x > 0 \tag{25}$$

where $H_{sep} = \left[n_i, -n_i\widehat{p_i}\right]$ for all separating contacts $m_{cs}^{(i)} = 1$, $H_c = \left[n_j, -n_j\widehat{p_j}\right]$ for all contacting contacts $m_{cs}^{(j)} = 0$. We have hyperplanes $H_{ss} = \left[T_j, -T_j\hat{p_j}\right]$ for all contacting contacts $m_{cs}^{(j)} = 0$. If we let $H_o = [H_{sep}, H_{ss}]$, all valid sliding/sticking modes can be written as

$$M_{ss} = \{\text{sgn}(H_o x) : x \in \mathbb{R}^d, H_{sep}x > 0, H_c x = 0\}. \tag{26}$$

Our algorithm computes the combinatorial structure of the hyperplane arrangement H_o to get the sliding/sticking modes M_{ss}.

Project to contact planes: We project H_o into halfspaces on the coordinate of the null space H_c to speed up computation. In Algorithm 2 step 3-5, the projected hyperplanes are obtained by $H = \left[H_{sep} \cdot \text{NULL}(H_c), H_{ss} \cdot \text{NULL}(H_c)\right]$.

Construct the zonotope: From the set of hyperplanes H, we can identify the vertices of its associated zonotope $Z(H)$. From Equation 17, zonotopes can be represented by the minkowski sum of line segments, which in this case are $\left[h_i, -h_i\right]$ for $h_i \in H$. Algorithm 2 Function GET-ZONOTOPE-VERTICES obtains zonotope vertices V through computing the minkowski sum iteratively for every $h_i \in H$. We initialize the vertex set $V = [0]$, and at the i-th iteration we update V with the convex hull of $\cup_{v \in V}\{v + h_i, v - h_i\}$. We are also able to get the sign vector for every vertex according to Equation 18: $\text{sgn}(v + h_i) = [\text{sgn}(v), +1]$, $\text{sgn}(v - h_i) = [\text{sgn}(v), -1]$.

Build the face lattice Using the same method as describe in Section 5.1, we construct the face lattice $L(V)$ from the vertices V. Not every $F \in L(V)$ corresponds to a valid sliding/sticking contact mode. Only the faces that have positive signs $+1$ with respect to normal velocity constraints for all separating contacts are valid sliding/sticking contact modes. After building the face lattice $L(V)$, valid faces $\mathcal{F}$ are selected by ensuring their sign vectors with respect to H_{sep} are all $\{+1\}$s:

$$\mathcal{F} = \{F \in L(V) : \text{sgn}_{H_{sep}}(F) = [+1, \ldots, +1]\} \tag{27}$$

The sign vectors of all faces in $\mathcal{F}$ with respect to H represent all valid sliding/sticking contact modes for the given contact/separating mode.

Theorem 2. *For a set of n contacts (modeled with k tangent planes) in a system of colliding bodies with d degrees of freedom, Algorithm 2 enumerates the possible sliding/sticking modes in $O(n^{d^2/2+2d})$ time for a given contacting/separating mode.*

Proof. As before, we first proceed with a proof of correctness. For a given contacting/separating mode string m_{cs}, let $H = [h_{s_1}, \cdots, h_{s_k}, h_{t_1}, \cdots, h_{t_m}]$ be the input hyperplanes to our zonotope construction algorithm, where k is the number of separating hyperplanes and m is the number of tangent hyperplanes. We incrementally construct the zonotope by using the fact that the Minkowski sum of two polytopes is the convex hull of the sums of their vertices [6]. By Corollary 7.18 of [27], the face lattice of the zonotope constructed above is the opposite of the face lattice of the hyperplane arrangement.

Next we analyze the complexity of our algorithm. The maximum number of hyperplanes is kn. The number of vertices, i.e. f_0, for a d-zonotope that is the projection of a p-cube is of the order $O(p^{d-1})$ [7]. Therefore, our zonotope construction algorithm takes time

$$O(\sum_{p=1}^{kn}(p^{d-1})^{\frac{d+1}{2}}) \approx O(\sum_{p=1}^{kn} p^{\frac{d^2}{2}}) \approx O((kn)^{\frac{d^2}{2}}). \tag{28}$$

We use Kaibel and Pfetsch [14] to construct the face lattice of the resulting zonotope. As before, their algorithm runs in

$$O(kn \cdot d \cdot \alpha \cdot (kn)^{d-1}) \approx O(d(kn)^{2d}), \tag{29}$$

where $\alpha = kn \cdot (kn)^{d-1}$ in the worst case (when the zonotope is simple) [27]. The full complexity of SS-ENUMERATE is therefore $O((kn)^{\frac{d^2}{2}+2d})$.

5.3 Full Mode Enumeration

Theorem 3. *For a set of n contacts (modeled with k tangent planes) in a system of colliding bodies with d degrees of freedom, Algorithm 1 and 2 can be combined to enumerate the full set of valid contact modes is in $O(n^{d^2/2+2.5d})$ time.*

Proof. From Theorem 1, we can enumerate contacting/separating modes in $O(d{\cdot}n^{d+1}+ l(n,d))$ and there are at most $O(n^{d/2})$ such modes. From Theorem 2, given a contacting/separating mode, we can enumerate sticking/sliding modes in $O((kn)^{\frac{d^2}{2}+2d})$. Therefore, running the full mode enumeration takes at most

$$O(d \cdot n^{d+1} + l(n,d) + n^{d/2} \cdot (kn)^{\frac{d^2}{2}+2d}) \approx O(n^{d^2/2+2.5d}) \tag{30}$$

Note that $O(n^{d^2/2+2.5d})$ is the worst-case time complexity of our algorithm, which only happens when the zonotope is simple [27]. Additionally, if we define sliding and sticking for nonlinear friction cones as any non-zero tangent velocity or zero tangent velocity, respectively, then it is clear our algorithm also enumerates modes for nonlinear friction cones.

6 Results

This section collects the results of running our algorithms (cs-mode-enum, ss-mode-enum, all-modes) on the example scenarios described below. The examples were run on a computer with an Intel i7-7820x CPU (3.5 MHz, 16 threads).

Box-#: This example simulates a box enclosed by # walls, $1 \leq \# \leq 5$. The face contact between the box and a wall is modeled as four point contacts on the corresponding vertices of the box. Constrained by one wall (the ground), the box is free to separate with or move along the wall. As the constraints grow to 6 walls, the box becomes progressively more constrained. Figure 2 and 3 visualize some of our results.

Peg-in-hole-#: This example consists of a cylindrical peg sitting within a cube with circular hole. The number # corresponds to the number of contact points generated at each end of the peg. To demonstrate the generality of our system, we simulate the peg

			cs-modes					ss-modes		
	n	d	d_{aff}	$\sum f_k$	t_{lp}	t_{conv}	t_{lattice}	$\sum f_k^{cs}$	t_{zono}	t_{lattice}
box-1	4	6	2	10	2.66e-3	9.68e-5	1.39e-4	196	1.77e-2	8.38e-3
box-2	8	6	4	46	3.25e-3	1.55e-4	1.69e-3	764	4.18e-2	3.73e-2
box-3	12	6	5	190	3.18e-3	2.72e-4	1.53e-2	2386	1.56e-1	1.17e-1
box-4	16	6	2	10	2.67e-3	8.94e-5	1.34e-4	12	4.20e-2	9.73e-4
box-5	20	6	1	2	2.75e-3	0	0	2	0	0
peg-in-hole-4	8	6	1	1	2.69e-3	0	0	9	4.60e-3	3.66e-4
peg-in-hole-8	16	6	1	1	2.89e-3	0	0	9	9.80e-3	3.66e-4
box-box-1	8	9	2	10	2.30e-3	8.56e-3	1.29e-4	9945	1.21e-1	2.22e+1
box-box-2	16	9	6	184	2.32e-3	1.99e-4	1.33e-2	168746	4.43e+0	9.87e+2
hand-ball	12	17	11	4096	4.08e-3	2.23e-4	7.01e-1	–	>1hr	>1hr

Table 1: Results of contact mode enumeration in 3D

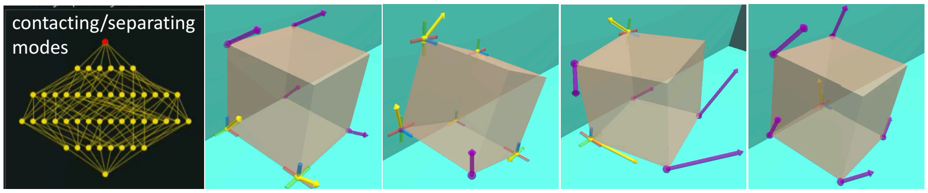


Fig. 2: Visualization of some contact modes of a box enclosed by 2 walls. Given a contact mode, we sampled object velocity and rendered it in the simulation. Yellow arrows: velocities of contacting contacts. Purple arrows: velocities of separating contacts.

with 6 degrees of freedom even though its constraints reduce the effective DoF of the system to 2 (rotating or translating about the main axis).

Box-box-#: This example consists of a box with 6 DoFs sitting on a box with 3 DoFs (translation in x and y and rotation about z). As before, # is the number of walls.

Hand-ball: This example simulates an anthropomorphic hand grabbing a football (ellipsoid). The hand has 3 fingers and a thumb, with 3 (2) DoFs per finger (thumb), and the football has 6 DoFs for a total of 17 DoFs in the system. The football is posititioned so that when grasped, it contacts every link on the hand, including the palm.

The above examples can be replicated ad infinitum (by adding more boxes or fingers) to generate contact modes which grow exponentially with the system dimension. Table 1 aggregates the results of running our algorithm on the above examples. Selected videos of the examples are also provided as supplementary material. From the results, it is clear that lattice construction takes the most percentage of time. In the future, we intend to implement methods from Edelsbrunner [7] which have better complexity.

7 Related Applications

Simulation: Friction contact dynamics have been modeled as complementarity problems by many researchers [2, 24]. Approaches for solving complementarity problems may be broken into direct methods, such as Lemke's algorithm, and iterative methods,

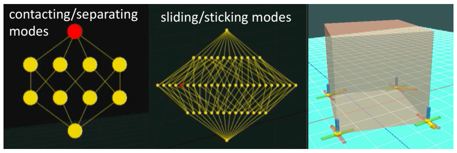


Fig. 3: The contacting/separating modes and sliding/sticking modes of a box on the table case.

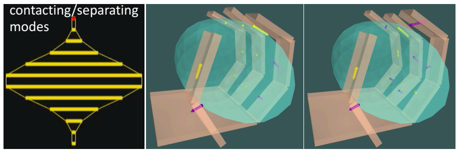


Fig. 4: A box in a hand. There are too many contacting/separating modes so we only visualize them by layers.

such as Projected Gauss Siedel [20, 5]. The simulation research community has focused on issues such as proving finite termination [26, 24], satisfying physical realism [23, 25], and improving convergence speed [15]. By design these algorithms find a single solution, when in reality, multiple solutions may exist [24]. Moreover, users are not aware of this additional source of divergence. However, our algorithm can allow the user to enumerate all possible solutions. We believe this capability will be important for robotic systems which use simulation to reason about future actions.

Grasping and Dexterous Manipulation: Most grasping synthesis algorithms [4] [19] [18] are designed to plan force-closure graspings. However, due to the static indeterminacy problem, a force closure grasp does not imply a stable grasp [17]. A typical way to address this problem is to enumerate all adjacent contact modes and make sure they don't have the same solution to the desired contact mode. Our 3D contact enumeration method could provide fast computation tools for stable grasps in 3D, which may help with real-time grasp planning for large scale objects. Similar approach can also be extended to dexterous manipulation tasks, like pushing [16] and grasping using environment contacts [12], where certain contact mode is desired during the task.

8 Conclusion

This paper introduced the first known algorithm for efficient contact mode enumeration in 3D. The algorithm partitions the problem into contacting/separating mode enumeration and sliding/sticking mode enumeration. The algorithms are based on convex hull and hyperplane arrangements, respectively. This paper also presented results demonstrating real-time enumeration for small problems. Finally, we highlighted related research areas to show that contact mode enumeration can be a useful tool for the simulation, analysis, and control of robotic systems which make and break contact with the environment.

Bibliography

[1] Avis D, Fukuda K (1996) Reverse search for enumeration. Discrete Applied Mathematics 65(1):21–46, DOI 10.1016/0166-218X(95)00026-N, URL http://www.sciencedirect.com/science/article/pii/0166218X9500026N

[2] Baraff D (1991) Coping with friction for non-penetrating rigid body simulation. In: Proceedings of the 18th annual conference on Computer graphics and interactive techniques, Association for Computing Machinery, New York, NY, USA, SIGGRAPH '91, pp 31–41, DOI 10.1145/122718.122722, URL https://doi.org/10.1145/122718.122722

[3] Barber CB, Dobkin DP, Huhdanpaa H (1996) The quickhull algorithm for convex hulls. ACM Trans Math Softw 22(4):469–483, DOI 10.1145/235815.235821, URL https://doi.org/10.1145/235815.235821

[4] Bicchi A (1995) On the closure properties of robotic grasping. The International Journal of Robotics Research 14(4):319–334

[5] Cottle RW, Pang JS, Stone RE (2009) The Linear Complementarity Problem. Classics in Applied Mathematics, Society for Industrial and Applied Mathematics, DOI 10.1137/1.9780898719000, URL https://epubs.siam.org/doi/abs/10.1137/1.9780898719000

[6] Delos V, Teissandier D (2015) Minkowski Sum of Polytopes Defined by Their Vertices. Journal of Applied Mathematics and Physics 03(01):62, DOI 10.4236/jamp.2015.31008, URL http://www.scirp.org/journal/PaperInformation.aspx?PaperID=53577&#abstract

[7] Edelsbrunner H (2005) Algorithms in Combinatorial Geometry. Springer-Verlag, Berlin, oCLC: 890383639

[8] Facchinei F, Pang JS (2007) Finite-dimensional variational inequalities and complementarity problems. Springer Science & Business Media

[9] Geyer T, Torrisi FD, Morari M (2010) Efficient mode enumeration of compositional hybrid systems. International Journal of Control 83(2):313–329, DOI 10.1080/00207170903159285, URL https://doi.org/10.1080/00207170903159285

[10] Greenfield A, Saranli U, Rizzi AA (2005) Solving Models of Controlled Dynamic Planar Rigid-Body Systems with Frictional Contact. The International Journal of Robotics Research 24(11):911–931, DOI 10.1177/0278364905059056, URL https://doi.org/10.1177/0278364905059056

[11] Haas-Heger M, Papadimitriou C, Yannakakis M, Iyengar G, Ciocarlie M (2018) Passive Static Equilibrium with Frictional Contacts and Application to Grasp Stability Analysis. In: Proceedings of Robotics: Science and Systems, Pittsburgh, Pennsylvania, DOI 10.15607/RSS.2018.XIV.064

[12] Hou Y, Jia Z, Mason MT (2020) Manipulation with shared grasping. Under Review

[13] Johnson AM, Burden SE, Koditschek DE (2016) A Hybrid Systems Model for Simple Manipulation and Self-Manipulation Systems. International Journal of Robotics Research 35(11):1354–1392

[14] Kaibel V, Pfetsch ME (2002) Computing the Face Lattice of a Polytope from its Vertex-Facet Incidences. arXiv:math/0106043 URL http://arxiv.org/abs/math/0106043, arXiv: math/0106043

[15] Lloyd J (2005) Fast Implementation of Lemke's Algorithm for Rigid Body Contact Simulation. In: Proceedings of the 2005 IEEE International Conference on Robotics and Automation, pp 4538–4543, DOI 10.1109/ROBOT.2005.1570819, iSSN: 1050-4729

[16] Lynch KM, Mason MT (1996) Stable pushing: Mechanics, controllability, and planning. The international journal of robotics research 15(6):533–556

[17] Mason MT (2001) Mechanics of robotic manipulation. MIT press

[18] Miller AT, Allen PK (2004) Graspit! a versatile simulator for robotic grasping. IEEE Robotics Automation Magazine 11:110–122

[19] Murray RM, Sastry SS, Ze-xiang L (1994) A mathematical introduction to robotic manipulation

[20] Murty KG, Yu FT (1988) Linear complementarity, linear and nonlinear programming, vol 3. Citeseer

[21] Potočnik B, Mušič G, Zupančič B (2004) A New Technique for Translating Discrete Hybrid Automata into Piecewise Affine Systems. Mathematical and Computer Modelling of Dynamical Systems 10(1):41–57, DOI 10.1080/13873950412331318062, URL https://doi.org/10.1080/13873950412331318062

[22] QHull Library (2020) QHalf Notes. URL http://www.qhull.org/html/qhalf.htm#notes

[23] Smith B, Kaufman DM, Vouga E, Tamstorf R, Grinspun E (2012) Reflections on Simultaneous Impact. ACM Transactions on Graphics (Proceedings of SIGGRAPH 2012) 31(4):106:1–106:12

[24] Stewart DE, Trinkle JC (1996) An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and coulomb friction. International Journal for Numerical Methods in Engineering 39(15):2673–2691

[25] Tonge R, Benevolenski F, Voroshilov A (2012) Mass splitting for jitter-free parallel rigid body simulation. ACM Transactions on Graphics (TOG) 31(4):105:1–105:8, DOI 10.1145/2185520.2185601, URL https://doi.org/10.1145/2185520.2185601

[26] Vouga E, Smith B, Kaufman DM, Tamstorf R, Grinspun E (2017) All's well that ends well: guaranteed resolution of simultaneous rigid body impact. ACM Transactions on Graphics (TOG) 36(4):151:1–151:19, DOI 10.1145/3072959.3073689, URL https://doi.org/10.1145/3072959.3073689

[27] Ziegler GM (1995) Lectures on Polytopes. Graduate Texts in Mathematics, Springer-Verlag, New York, DOI 10.1007/978-1-4613-8431-1, URL https://www.springer.com/gp/book/9780387943299