Chapter 6

Local Motion Constraints

In this chapter, the basic motion constraint that prevents inter-penetration from occurring near contact points is derived. The motion constraint at one contact point does not prevent inter-penetration at another contact point, and additional motion constraints are required. For this reason, the motion constraint at each contact point is said to be a *local* constraint, as it is responsible for preventing inter-penetration from occurring only in the immediate vicinity of the contact point, and only over some (possibly small) range of motion. This locality property, and the fact that the constraint is meant to apply only over some small range of motion is to be implicitly assumed in the rest of our discussion.

We will start by expressing the non-penetration constraint between bodies in terms of the spatial variables of the contacting bodies. By differentiating this constraint twice with respect to time, a constraint on the accelerations of the bodies is produced. Chapter 8 will show how workless constraint forces can be computed that satisfy this acceleration constraint for each contact point. The derivation of

these acceleration constraints is well known for certain special cases of contact, but as far as we know a formulation for the general case has been lacking in the literature.

6.1 Configuration Space

The problem of preventing inter-penetration at any one contact point between two bodies can be viewed as a more complicated version of the simple particle/surface problem of Chapter 2, by considering the contact in terms of the *configuration space*, or C-space, of the two bodies[1,12,31,32]. The C-space of a rigid body is a six-dimensional space, representing the degrees of freedom of the body. A single point in this C-space is specified by six coordinates; each separate coordinate represents one degree of freedom of the body. Thus, each point of this C-space represents a particular configuration (a position and orientation) of the rigid body. The C-space of a pair of rigid bodies is 12 dimensional; a single point of this C-space represents a configuration for each body. We will describe the motion constraint between two bodies at a single contact point using this 12 dimensional C-space.

Points in C-space will be denoted by a horizontal bar, such as $\overline{p}$, while points in world space ($\mathbf{R}^3$) will be written simply as p. Both points p in $\mathbf{R}^3$ and points $\overline{p}$ in C-space denote column vectors. Partial derivative operators $\partial/\partial p$ and $\partial/\partial \overline{p}$ yield column vectors when applied to functions.

At time t_0, consider two bodies A and B that contact at p_c, and whose configuration is represented by the point $\overline{x}_0$. By considering the contact in C-space, we can employ the standard technique of viewing the constraint that A and B remain in contact near p_c as a manifold S, containing $\overline{x}_0$, in some neighborhood of $\overline{x}_0$. The set of all points in C-space near $\overline{x}_0$ that represent configurations such that A and B remain in contact (near p_c) is a manifold, S, containing $\overline{x}_0$. Moreover S is a "hypersurface" in C-space; that is, S has an 11-dimensional tangent space

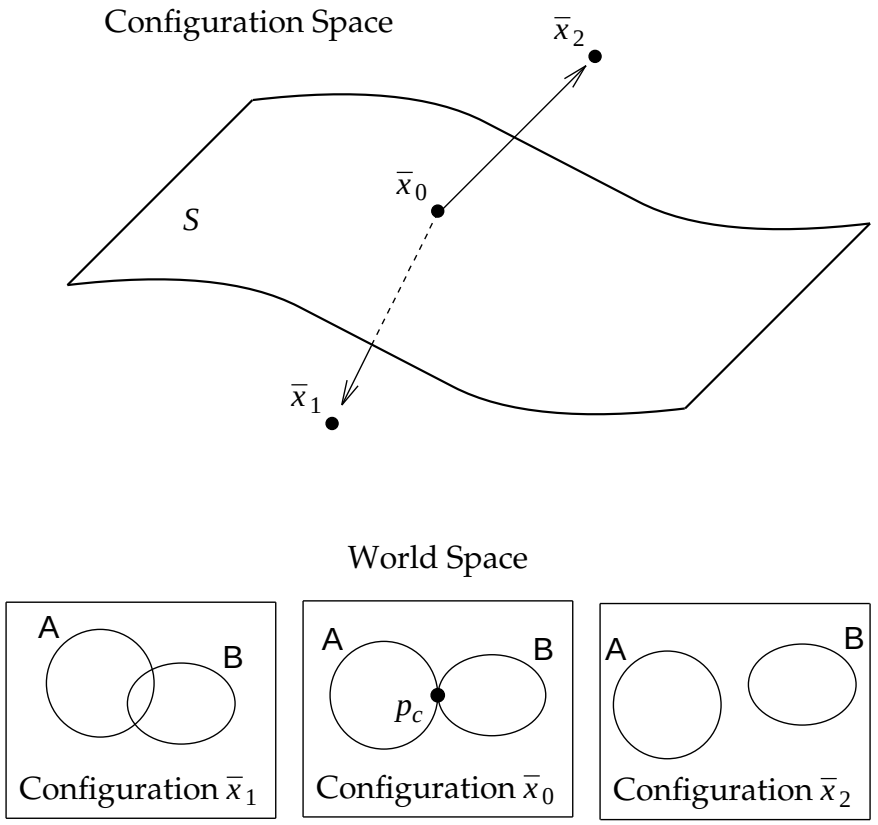

Figure 6.1: Three configurations in world space and their representation as points in C-space. The points $\overline{x}_1$ and $\overline{x}_2$ lie on opposite sides of the hypersurface S.

and a single normal direction at each point.

The hypersurface S partitions C-space near $\overline{x}_0$ into two regions. Points near $\overline{x}_0$ on one side of S represent configurations where A and B are disjoint near p_c; points near $\overline{x}_0$ on the other side represent configurations where A and B have inter-penetrated near p_c (Figure 6.1).

Since we are interested in the behavior of A and B over time, let $\overline{x}(t)$ be the configuration of A and B at time t. Then $\overline{x}_0 = \overline{x}(t_0)$. We will describe S implicitly in terms of a scalar function C of C-space; that is,

$$S = \{\, \overline{p} \mid C(\overline{p}) = 0 \,\}. \tag{6–1}$$

We will adopt the convention that if $C(\overline{p}) < 0$, then $\overline{p}$ represents a configuration such that A and B inter-penetrate near p_c. Conversely, if $C(\overline{p}) > 0$, then $\overline{p}$ represents a configuration such that A and B are disjoint near p_c. Since A and B contact at p_c at time t_0, we know that

$$C(\overline{x}(t_0)) = 0. \tag{6–2}$$

In Figure 6.1, C is positive for configurations lying "above" S; thus, valid (non-penetrating) configurations are those that lie on or "above" S.

Given the above definitions, the C-space representation of the non-penetration constraint for an arbitrary contact has the same form as the non-penetration constraint for the point/surface problem of Section 2.3. To prevent inter-penetration, $\overline{x}(t)$ must be prevented from lying below S; thus

$$C(\overline{x}(t)) \geq 0 \tag{6–3}$$

enforces the non-penetration constraint. Comparing this equation with Equation (2–4), the analogy between the particle/surface non-penetration constraint and the non-penetration constraint for an arbitrary contact is obvious.

6.2 Projections onto S

Equation (6–3) expresses the non-penetration constraint in terms of spatial variables. We convert the non-penetration constraint into a constraint on acceleration by differentiating. Consider

$$\dot{C}(\overline{x}(t)) = \frac{\mathrm{d}}{\mathrm{dt}} C(\overline{x}(t)).$$

The function $\dot{C}(\overline{x}(t_0))$ measures relative velocity in the same manner as $\dot{C}(x(t_0))$ of Equation (2–10). If $\dot{C}(\overline{x}(t_0)) < 0$, then A and B are colliding (near p_c), while if

$\dot{C}(\overline{x}(t_0)) > 0$ then A and B are breaking contact at p_c; otherwise $\dot{C}(\overline{x}(t_0)) = 0$ and A and B are neither colliding nor separating at p_c. Since collisions are assumed to have been dealt with prior to this stage, and the non-penetration constraint would not be active if A and B were separating at p_c, we can assume $\dot{C}(\overline{x}(t_0)) = 0$.

Since $C(\overline{x}(t_0)) = \dot{C}(\overline{x}(t_0)) = 0$, inter-penetration is prevented by the condition

$$\ddot{C}(\overline{x}(t_0)) = \frac{\mathrm{d}^2}{\mathrm{dt}^2} C(\overline{x}(t_0)) \geq 0. \tag{6–4}$$

The quantity $\ddot{C}(\overline{x}(t_0))$ is a measure of the *acceleration* between A and B at p_c, in the normal direction. If $\ddot{C}(\overline{x}(t_0))$ is positive, then A and B are breaking contact, while if $\ddot{C}(\overline{x}(t_0))$ is zero, then A and B are remaining in contact.

Given any net forces and torques applied to A and B, these forces and torques can be translated into a net C-space applied force acting on the configuration point $\overline{x}_0$. If the contact between A and B is frictionless, then a C-space reaction force normal to the hypersurface S at $\overline{x}_0$ arises (if necessary) to prevent $\overline{x}(t)$ from moving downward.[1] The C-space reaction force is computed by projecting the C-space applied force onto the C-space normal, and reversing its direction (Figure 6.2).

Thus, it is assumed by those accustomed to working in C-space that computing the reaction force, in either C-space or world space, is a simple matter. However, the computation of the net C-space applied force can be difficult, because it must take into account inertial forces generated by the motion of A and B. Consider $\dot{C}(\overline{x}(t_0))$. Applying the chain rule,

$$\frac{\mathrm{d}}{\mathrm{dt}} C(\overline{x}(t_0)) = \frac{\partial C}{\partial \overline{p}} \left(\overline{x}(t_0)\right)^T \dot{\overline{x}}(t_0). \tag{6–5}$$

[1] The normal direction is defined by the the natural inner product induced by the kinetic energy of the system. The projection of Figure 6.2 must also be performed using the natural inner product, but the intuition behind Figure 6.2 is valid. Erdmann[12] contains a more detailed discussion of the subject.

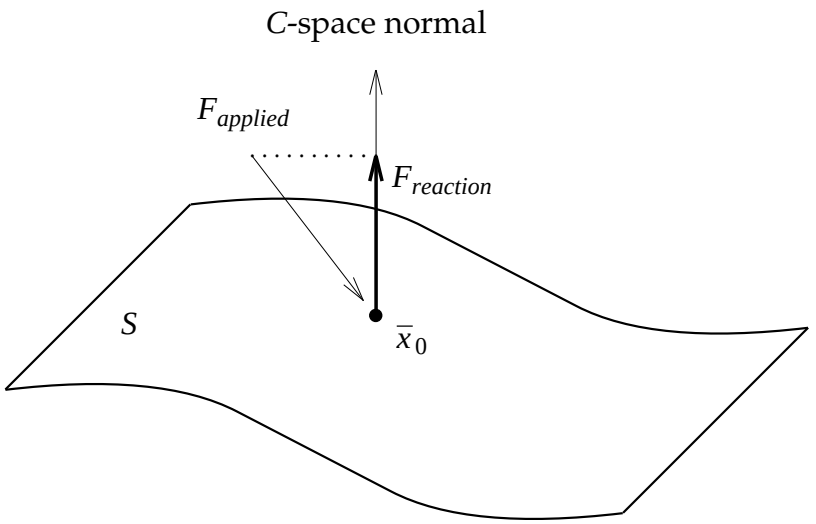

Figure 6.2: The reaction force $F_{reaction}$. The reaction force is computed by projecting the applied force $F_{applied}$ onto the C-space normal of S.

(Recall that both $\partial C/\partial \overline{p}$ and $\dot{\overline{x}}$ are column vectors.) Differentiating again,

$$\begin{aligned}\frac{\mathrm{d}^2}{\mathrm{d}t^2}C(\overline{x}(t_0)) &= \left(\frac{\partial}{\partial \overline{p}}\frac{\partial C}{\partial \overline{p}}(\overline{x}(t_0))\dot{\overline{x}}(t_0)\right)^T \dot{\overline{x}}(t_0) + \frac{\partial C}{\partial \overline{p}}(\overline{x}(t_0))^T \ddot{\overline{x}}(t_0) \\ &= \dot{\overline{x}}(t_0)^T \frac{\partial^2 C}{\partial \overline{p}^2}(\overline{x}(t_0))\dot{\overline{x}}(t_0) + \frac{\partial C}{\partial \overline{p}}(\overline{x}(t_0))^T \ddot{\overline{x}}(t_0) \qquad (6\text{–}6)\end{aligned}$$

with

$$\frac{\partial^2 C}{\partial \overline{p}^2}(\overline{x}(t_0))$$

the (symmetric) 12×12 matrix of second partial derivatives of C at $\overline{x}(t_0)$. The right term of the last sum in Equation (6–6) is the component of acceleration of $\overline{x}(t_0)$ normal to the C-space surface S. The left term of the last sum represents the inertial forces acting on A and B. In order to evaluate this term it is necessary to calculate the second partial derivatives of C at $\overline{x}_0$. This has been done only for the case of polyhedral contact[12], but appears not to have been done for the more general case of contact between curved surfaces. Thus, the computation of the

reaction force when either there is no initial motion ($\dot{\bar{x}}(t_0) = 0$) or the contacting bodies are polyhedral is simply accomplished. In the next section, the second partial derivatives of C are computed for the more general case of contacting surfaces in motion. The derivation for the second partial derivatives of C yields a result that is not particularly tractable for use in a simulation environment. As a result, in Section 6.4, a more tractable expression for Equation (6–6) is derived, that does not explicitly involve the computation of the second partial derivatives of C. However, an analytical expression for the second partial derivatives of C may be useful in some contexts, so a suitable expression is derived in the next section.

6.3 Calculating $\partial^2 C / \partial \overline{p}^2$

In order to create a constructive definition of C for curved surfaces, we will use the concepts of the extremal points between two curved surfaces, from Section 5.2.1. Since surfaces need not be convex in this chapter, the extremal points p_a and p_b can only be defined locally, in the vicinity of a contact point. The extremal points near p_c are defined as follows. If A and B are disjoint near p_c, then the extremal points minimize the distance between A and B near p_c. If A and B are in contact at p_c, then the extremal points are $p_a = p_b = p_c$. If A and B have inter-penetrated near p_c, then the extremal points maximize the distance distance between A and B (near p_c). The points p_a and p_b at a contact point vary according to the configuration $\overline{p}$ of A and B; thus, p_a and p_b may be considered as functions of configuration (Figure 6.3).

Again, as in Section 5.2, we will model the surface of A and B near p_c implicitly by functions F and G; however, F and G will now be functions of *configuration* as opposed to time. That is, if the configuration of A and B is $\overline{p}$, then a point p is on the surface of A if $F(p, \overline{p}) = 0$, and similarly for B. Again, A and B can be

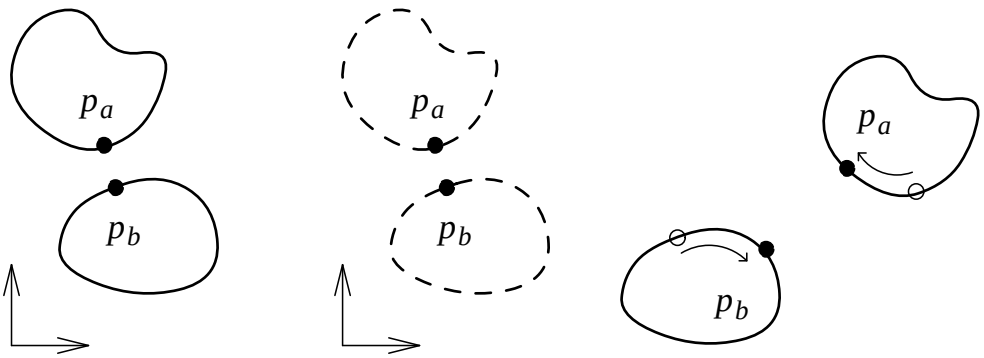

Figure 6.3: Change of the extremal points as a function of the configuration.

modeled equally well by parametric functions; Appendix A details the derivations for the parametric case. Partial derivatives of F and G with respect to $\overline{p}$ are easily computed, assuming partial derivatives of F and G with respect to p are known in some rest frame.

As in Section 5.2, the functions F and G are such that the surface normal of F,

$$\nabla F = \frac{\partial F}{\partial p},$$

is outwards pointing, and similarly for G. Thus, the function C can be written simply as

$$C(\overline{p}) = \nabla G(p_b, \overline{p}) \cdot (p_a - p_b) \tag{6–7}$$

where p_a and p_b are the extremal points for the configuration $\overline{p}$ (Figure 6.4). Note that $\nabla G(p_b, \overline{p})$ is always parallel to the vector $p_a - p_b$. Differentiating,

$$\frac{\partial C}{\partial \overline{p}}(\overline{p}) = \frac{\partial}{\partial \overline{p}} \nabla G(p_b, \overline{p}) \cdot (p_a - p_b) + \nabla G(p_b, \overline{p}) \cdot \left(\frac{\partial p_a}{\partial \overline{p}} - \frac{\partial p_b}{\partial \overline{p}} \right). \tag{6–8}$$

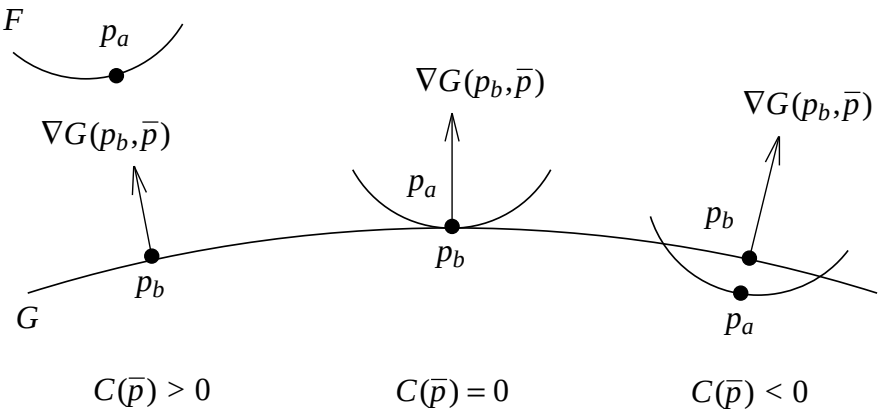

Figure 6.4: C expressed in terms of the extremal points.

Then

$$\begin{aligned}
\frac{\partial^2 C}{\partial \overline{p}^2}(\overline{p}) &= \frac{\partial}{\partial \overline{p}}\frac{\partial C}{\partial \overline{p}}(\overline{p}) \\
&= \frac{\partial^2}{\partial \overline{p}^2}\nabla G(p_b,\overline{p}) \cdot (p_a - p_b) + \frac{\partial}{\partial \overline{p}}\nabla G(p_b,\overline{p}) \cdot \left(\frac{\partial p_a}{\partial \overline{p}} - \frac{\partial p_b}{\partial \overline{p}}\right) \\
&\quad + \frac{\partial}{\partial \overline{p}}\nabla G(p_b,\overline{p}) \cdot \left(\frac{\partial p_a}{\partial \overline{p}} - \frac{\partial p_b}{\partial \overline{p}}\right) + \nabla G(p_b,\overline{p}) \cdot \left(\frac{\partial p_a}{\partial \overline{p}^2} - \frac{\partial p_b}{\partial \overline{p}^2}\right) \quad (6\text{–}9) \\
&= \frac{\partial^2}{\partial \overline{p}^2}\nabla G(p_b,\overline{p}) \cdot (p_a - p_b) + 2\frac{\partial}{\partial \overline{p}}\nabla G(p_b,\overline{p}) \cdot \left(\frac{\partial p_a}{\partial \overline{p}} - \frac{\partial p_b}{\partial \overline{p}}\right) \\
&\quad + \nabla G(p_b,\overline{p}) \cdot \left(\frac{\partial p_a}{\partial \overline{p}^2} - \frac{\partial p_b}{\partial \overline{p}^2}\right).
\end{aligned}$$

Clearly, to evaluate the second partial derivatives of C, we must be able to differentiate p_a and p_b in terms of the configuration $\overline{p}$. This requires a more constructive definition for p_a and p_b.

In Section 5.2.1, a set of necessary conditions was given for the extremal points. Because we are defining p_a and p_b only locally in this chapter, the conditions of Section 5.2.1 are both necessary and sufficient. That is, if we say that p_a and p_b

are the points near p_c that satisfy the conditions

$$\begin{cases} E_1: & \nabla F(p_a, \overline{p}) + \lambda_2 \nabla G(p_b, \overline{p}) = \mathbf{0} \\ E_2: & F(p_a, \overline{p}) = 0 \\ E_3: & G(p_b, \overline{p}) = 0 \\ E_4: & (p_b - p_a) + \lambda_1 \nabla G(p_b, \overline{p}) = \mathbf{0} \end{cases} \tag{6–10}$$

then this uniquely defines p_a and p_b in some neighborhood of p_c for a given configuration $\overline{p}$. Equation (6–10) is modified exactly as in Section 5.2.2 for the case of contact between a polyhedron and a curved surface. Constraint functions for contact between two polyhedra are easily formulated[12,13] if the contact involves either two non-parallel contacting edges, or a vertex in contact with the interior of a polygonal face; both these cases admit a well defined normal direction. However, contact between two vertices, a vertex and an edge, or two parallel edges do not yield a well defined normal and may be considered as *degenerate* contact situations. Palmer[42] discusses the complexity of simulations involving degenerate contacts; in general, degenerate contacts yield *NP*-hard simulation problems. We will assume that a normal direction is defined at all contact points.

We can use Equation (6–10) to differentiate p_a and p_b with respect to $\overline{p}$. Suppose we are given values for p_a and p_b for some configuration $\overline{p}$. (Typically, we wish to differentiate p_a and p_b for the configuration $\overline{x}_0$; for this configuration, $p_a = p_b = p_c$.) Given p_a and p_b, values for λ_1 and λ_2 are easily calculated. Using Equation (6–10), we can calculate $\partial p_a / \partial \overline{p}$ and $\partial p_b / \partial \overline{p}$ by making use of the implicit function theorem for simultaneous equations[49]. This theorem asserts that under proper conditions, p_a and p_b may be regarded as functions of the configuration $\overline{p}$; the theorem also gives a constructive expression for the derivatives of p_a and p_b. We will write the Jacobian determinant of a set of vector functions $H_1(x_1, \ldots, x_n)$

through $H_n(x_1, \ldots, x_n)$ as

$$\frac{\partial(H_1, \ldots, H_n)}{\partial(x_1, \ldots, x_n)} = \begin{vmatrix} \dfrac{\partial H_1}{\partial x_1} & \cdots & \dfrac{\partial H_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial H_n}{\partial x_1} & \cdots & \dfrac{\partial H_n}{\partial x_n} \end{vmatrix}.$$

Since p_a and p_b are points in $\mathbf{R}^3$, p_a and p_b are each three (scalar) functions of configuration: $p_{ax}(\overline{p})$, $p_{ay}(\overline{p})$ and $p_{az}(\overline{p})$ and similarly for p_b. The implicit function theorem, applied to Equation (6–10) yields

$$\frac{\partial p_{ax}}{\partial \overline{p}_i}(\overline{p}) = -\frac{K}{J} \tag{6–11}$$

where

$$\begin{aligned} K &= \frac{\partial(E_1, E_2, E_3, E_4)}{\partial(\overline{p}_i, p_{ay}, p_{az}, p_{bx}, p_{by}, p_{bz}, \lambda_1, \lambda_2)} \\ J &= \frac{\partial(E_1, E_2, E_3, E_4)}{\partial(p_{ax}, p_{ay}, p_{az}, p_{bx}, p_{by}, p_{bz}, \lambda_1, \lambda_2)} \end{aligned} \tag{6–12}$$

and $\overline{p}_i$ denotes the ith component of $\overline{p}$. Similar expressions give the derivatives for p_{ay}, p_{az}, p_{bx}, p_{by} and p_{bz}. The case when J is near-zero or zero is considered in Section 6.7.

Second partial derivatives for p_a and p_b can be calculated by differentiating Equation (6–11). For example,

$$\frac{\partial^2 p_{ax}}{\partial \overline{p}_i \, \partial \overline{p}_j}(\overline{p}) = -\frac{\partial}{\partial \overline{p}_j}\left(\frac{K}{J}\right). \tag{6–13}$$

Combining this result with Equation (6–9) yields an analytical expression for the second partial derivatives of C. Examining the dimensions of the equations involved however, both K and J are the determinants of 8×8 matrices. Differentiating K and J with respect to $\overline{p}$ requires computing the derivative of an 8×8 matrix determinant. This is extremely impractical. Although K and J have considerable block

structure, block structure cannot be exploited in computing matrix determinants. In its present form, the Jacobian determinants of K and J contain more than 1,000 terms of seven factors each; the derivatives contain far more terms. Additionally, each of the 72 different second partials of p_{ax}, p_{ay}, p_{az}, p_{bx}, p_{by}, and p_{bz} would need to be computed in this manner. Thus, although we have achieved an analytical expression for Equation (6–6), it is not practical.

6.4 Computing $\ddot{C}(\overline{x}(t_0))$ directly

In this section, we shall formulate C in a slightly different manner. Let us assume a rotated coordinate system in which $\nabla F(p_a, \overline{x}(t_0))$ and $\nabla G(p_b, \overline{x}(t_0))$ are colinear with the z axis, with $\nabla G(p_b, \overline{x}(t_0))$ directed positively along z. (We will employ the standard right-handed coordinate system used to depict functions $z = h(x, y)$, with z the vertical axis.) Next, we explicitly model A and B near p_c as scalar functions f and g of x and y and configuration $\overline{p}$. Where F was a function $F(x, y, z, \overline{p})$, f is a function $f(x, y, \overline{p})$ and similarly for g. The justification for the existence of f and g is the implicit function theorem. The formal definitions of f and g are given by

$$F(x, y, f(x, y, \overline{p}), \overline{p}) = 0 \qquad \text{and} \qquad G(x, y, g(x, y, \overline{p}), \overline{p}) = 0. \tag{6–14}$$

Thus, the point $(x, y, f(x, y, \overline{x}(t)))$ is a point on F at time t, and similarly for G. In this new coordinate system, the extremal points p_a and p_b share the same x and y coordinates at time t_0. At times t near t_0, A and B do not penetrate as long as the z value of p_a is greater or equal to the z value of p_b (Figure 6.5). We will use this to redefine C as

$$C(\overline{p}) = f(p_{ax}, p_{ay}, \overline{p}) - g(p_{bx}, p_{by}, \overline{p}) \tag{6–15}$$

where p_{ax}, p_{ay}, p_{bx}, and p_{by} are the extremal points for configuration $\overline{p}$.

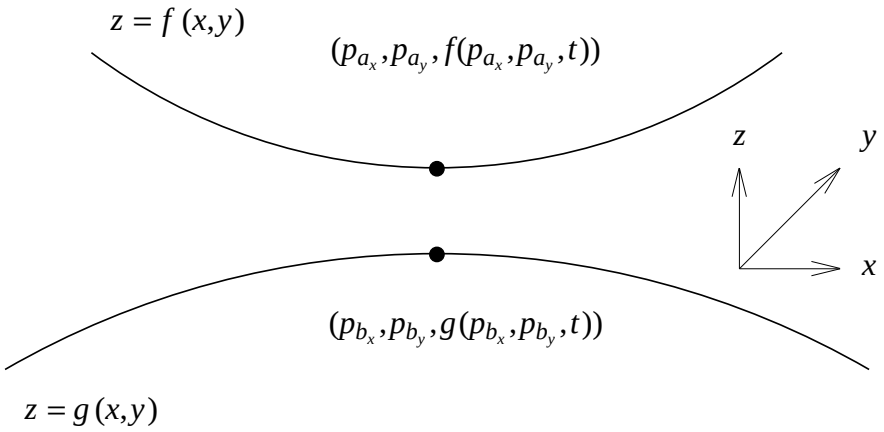

Figure 6.5: Side view of the implicit surfaces F and G expressed explicitly by functions f and g.

Since the z coordinates of the extremal points are no longer needed, let us rewrite Equation (6–10). Differentiating Equation (6–14) we obtain

$$\frac{\partial f}{\partial x} = -\frac{\dfrac{\partial F}{\partial x}}{\dfrac{\partial F}{\partial z}}, \quad \frac{\partial f}{\partial y} = -\frac{\dfrac{\partial F}{\partial y}}{\dfrac{\partial F}{\partial z}}, \qquad \text{and} \qquad \frac{\partial f}{\partial \overline{p}} = -\frac{\dfrac{\partial F}{\partial \overline{p}}}{\dfrac{\partial F}{\partial z}} \tag{6–16}$$

and similarly for g. Second derivatives of f and g are obtained by repeated differentiation of Equation (6–14). Using f and g, condition E_4 of Equation (6–10) may be written componentwise as

$$\begin{pmatrix} p_{bx} \\ p_{by} \\ g(p_{bx}, p_{by}, \overline{p}) \end{pmatrix} - \begin{pmatrix} p_{ax} \\ p_{ay} \\ f(p_{ax}, p_{ay}, \overline{p}) \end{pmatrix} + \lambda_1 \begin{pmatrix} \dfrac{\partial G}{\partial x}(p_b, \overline{p}) \\ \dfrac{\partial G}{\partial y}(p_b, \overline{p}) \\ \dfrac{\partial G}{\partial z}(p_b, \overline{p}) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \tag{6–17}$$

from which we obtain

$$\lambda_1 = \frac{f(p_{ax}, p_{ay}, \overline{p}) - g(p_{bx}, p_{by}, \overline{p})}{\dfrac{\partial G}{\partial z}(p_b, \overline{p})}. \tag{6–18}$$

Multiplying Equation (6–17) by -1 and using Equation (6–16) allows us to rewrite E_4 as

$$\begin{pmatrix} p_{ax} - p_{bx} \\ p_{ay} - p_{by} \end{pmatrix} + (f(p_{ax}, p_{ay}, \overline{p}) - g(p_{bx}, p_{by}, \overline{p})) \begin{pmatrix} \frac{\partial g}{\partial x}(p_{bx}, p_{by}, \overline{p}) \\ \frac{\partial g}{\partial y}(p_{bx}, p_{by}, \overline{p}) \end{pmatrix} = \mathbf{0}. \quad (6\text{–}19)$$

This new condition has one less equation than E_4 because of the reduction of variables from F to f. In a similar fashion, λ_2 of condition E_1 is eliminated and condition E_4 is rewritten as

$$\begin{pmatrix} \frac{\partial f}{\partial x}(p_{ax}, p_{ay}, \overline{p}) \\ \frac{\partial f}{\partial y}(p_{ax}, p_{ay}, \overline{p}) \end{pmatrix} - \begin{pmatrix} \frac{\partial g}{\partial x}(p_{bx}, p_{by}, \overline{p}) \\ \frac{\partial g}{\partial y}(p_{bx}, p_{by}, \overline{p}) \end{pmatrix} = \mathbf{0}. \quad (6\text{–}20)$$

Conditions E_2 and E_3 are no longer required since f and g give explicit definitions of the surfaces. At this point, it will be more useful to treat the extremal points as functions of time. We can define p_{ax}, p_{ay}, p_{bx} and p_{by} as the extremal points at time t by the system

$$\begin{cases} D_1: & \begin{pmatrix} \frac{\partial f}{\partial x}(p_{ax}, p_{ay}, \overline{x}(t)) \\ \frac{\partial f}{\partial y}(p_{ax}, p_{ay}, \overline{x}(t)) \end{pmatrix} - \begin{pmatrix} \frac{\partial g}{\partial x}(p_{bx}, p_{by}, \overline{x}(t)) \\ \frac{\partial g}{\partial y}(p_{bx}, p_{by}, \overline{x}(t)) \end{pmatrix} = \mathbf{0} \\ D_2: & (f(p_{ax}, p_{ay}, \overline{x}(t)) - g(p_{bx}, p_{by}, \overline{x}(t))) \begin{pmatrix} \frac{\partial g}{\partial x}(p_{bx}, p_{by}, \overline{x}(t)) \\ \frac{\partial g}{\partial y}(p_{bx}, p_{by}, \overline{x}(t)) \end{pmatrix} \\ & \qquad + \begin{pmatrix} p_{ax} - p_{bx} \\ p_{ay} - p_{by} \end{pmatrix} = \mathbf{0}. \end{cases} \quad (6\text{–}21)$$

Now p_{ax}, p_{ay}, p_{bx} and p_{by} can be differentiated as functions of time. The implicit function theorem for simultaneous equations gives the result

$$\dot{p}_{ax} = -\frac{\dfrac{\partial(D_1, D_2)}{\partial(t, p_{ay}, p_{bx}, p_{by})}}{J} \qquad \text{and} \qquad \dot{p}_{ay} = -\frac{\dfrac{\partial(D_1, D_2)}{\partial(p_{ax}, t, p_{bx}, p_{by})}}{J} \tag{6–22}$$

where

$$J = \frac{\partial(D_1, D_2)}{\partial(p_{ax}, p_{ay}, p_{bx}, p_{by})} \tag{6–23}$$

and

$$\frac{\partial f}{\partial t}(p_{ax}, p_{ay}, \overline{x}(t)) = \frac{\partial f}{\partial \overline{p}}(p_{ax}, p_{ay}, \overline{x}(t))^T \dot{\overline{x}}(t) \tag{6–24}$$

(and similarly for g). Similar expressions are obtained for $\dot{p}_{bx}$ and $\dot{p}_{by}$. The derivatives of p_{az} and p_{bz} are simply $\partial f/\partial t$ and $\partial g/\partial t$. Equation (6–23) redefines J, but we will not have any further use for the 8×8 Jacobian matrix in Equation (6–12). We will consider the case when J is near-zero or zero in Section 6.7.

Since D_1 and D_2 do not involve p_{az} or p_{bz}, let p_a and p_b henceforth denote the column vectors $(p_{ax}, p_{ay})^T$ and $(p_{bx}, p_{by})^T$ at time t. Similarly, let $\dot{p}_a = (\dot{p}_{ax}, \dot{p}_{ay})^T$ and $\dot{p}_b = (\dot{p}_{bx}, \dot{p}_{by})^T$ be the column vector derivatives of p_a and p_b at time t. The quantities $\partial f/\partial p$, $\partial f/\partial \overline{p}$ and $\dot{\overline{x}}(t)$ are also column vectors. From Equation (6–15), the condition $\ddot{C}(\overline{x}(t_0)) \geq 0$ is simply

$$\frac{\mathrm{d}^2}{\mathrm{dt}^2} f(p_a, \overline{x}(t_0)) - \frac{\mathrm{d}^2}{\mathrm{dt}^2} g(p_b, \overline{x}(t_0)) \geq 0. \tag{6–25}$$

Differentiating f with respect to time,

$$\frac{\mathrm{d}}{\mathrm{dt}} f(p_a, \overline{x}(t_0)) = \frac{\partial f}{\partial p}(p_a, \overline{x}(t_0))^T \dot{p}_a + \frac{\partial f}{\partial \overline{p}}(p_a, \overline{x}(t_0))^T \dot{\overline{x}}(t_0). \tag{6–26}$$

Then

$$\begin{aligned}
\frac{\mathrm{d}^2}{\mathrm{dt}^2} f(p_a, \overline{x}(t_0)) &= \left(\frac{\partial f}{\partial p^2}(p_a, \overline{x}(t_0))\dot{p}_a + \frac{\partial^2 f}{\partial p \partial \overline{p}}(p_a, \overline{x}(t_0))\dot{\overline{x}}(t_0) \right)^T \dot{p}_a \\
&\quad + \frac{\partial f}{\partial p}(p_a, \overline{x}(t_0))\ddot{p}_a \\
&\quad + \left(\frac{\partial^2 f}{\partial p \partial \overline{p}}(p_a, \overline{x}(t_0))\dot{p}_a + \frac{\partial f}{\partial \overline{p}^2}(p_a, \overline{x}(t_0))\dot{\overline{x}}(t_0) \right)^T \dot{\overline{x}}(t_0) \\
&\quad + \frac{\partial f}{\partial \overline{p}}(p_a, \overline{x}(t_0))^T \ddot{\overline{x}}(t_0) \qquad (6\text{–}27) \\
&= \dot{p}_a^T \frac{\partial f}{\partial p^2}(p_x, \overline{x}(t_0))\dot{p}_a + 2\dot{\overline{x}}(t_0)^T \frac{\partial^2 f}{\partial p \partial \overline{p}}(p_a, \overline{x}(t_0))\dot{p}_a \\
&\quad + \frac{\partial f}{\partial p}(p_a, \overline{x}(t_0))^T \ddot{p}_a + \dot{\overline{x}}(t_0)^T \frac{\partial f}{\partial \overline{p}^2}\dot{\overline{x}}(t_0) \\
&\quad + \frac{\partial f}{\partial \overline{p}}(p_a, \overline{x}(t_0))^T \ddot{\overline{x}}(t_0).
\end{aligned}$$

From Equation (6–16) and the fact that $\nabla F(p_{ax}, p_{ay}, \overline{x}(t_0))$ is colinear with the z axis,

$$\frac{\partial f}{\partial p}(p_a, \overline{x}(t_0)) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}. \qquad (6\text{–}28)$$

This yields

$$\begin{aligned}
\frac{\mathrm{d}^2}{\mathrm{dt}^2} f(p_a, \overline{x}(t_0)) &= \dot{p}_a^T \frac{\partial f}{\partial p^2}(p_a, \overline{x}(t_0))\dot{p}_a + 2\dot{\overline{x}}(t_0)^T \frac{\partial^2 f}{\partial p \partial \overline{p}}(p_a, \overline{x}(t_0))\dot{p}_a \\
&\quad + \dot{\overline{x}}(t_0)^T \frac{\partial f}{\partial \overline{p}^2}\dot{\overline{x}}(t_0) + \frac{\partial f}{\partial \overline{p}}\ddot{\overline{x}}(t_0). \qquad (6\text{–}29)
\end{aligned}$$

A similar expression is obtained for g. Thus, neither $\ddot{p}_a$ nor $\ddot{p}_b$ is required to calculate $\ddot{C}(\overline{x}(t_0))$. Equation (6–29) depends linearly on $\ddot{\overline{x}}(t_0)$, which in turn depends linearly upon the forces acting on A and B; this equation can be used to calculate the reaction force between A and B. Since $\dot{p}_a$ and $\dot{p}_b$ are simply calculated as the ratio of two 4×4 matrix determinants, the final expression is quite practical for use.

6.5 Predicting collision times

The primary use for the derivation of $\ddot{C}(\overline{x}(t_0))$ is to formulate the equations for the constraint forces at contact points. However, $\ddot{C}(\overline{x}(t_0))$ can also be used to help predict the time of a collision. When the collision/contact determination step detects inter-penetration between two bodies, an estimate must be made of when the two bodies collided. The work done by the simulator can be significantly decreased by increasing the accuracy of the prediction of the collision time.

Consider the case when inter-penetration is detected between two bodies at time t_1. Let $C(\overline{x}(t))$ represent the non-penetration constraint that has been violated in the vicinity of the inter-penetration.[2] Since the constraint is not satisfied at time t_1, we have $C(\overline{x}(t_1)) < 0$. However, if the previous time step (before the inter-penetration) occurred at time t_0, then $C(\overline{x}(t_0)) > 0$. Computing the collision time is thus a root finding problem, on some known interval: we are looking for a time t_c, with $t_0 < t_c < t_1$ such that $C(\overline{x}(t_c)) = 0$.

Since we can easily calculate derivatives of first and second derivatives of C, we model $C(\overline{x}(t))$ quadratically near t_0 as

$$C(\overline{x}(t)) = C(\overline{x}(t_0)) + \dot{C}(\overline{x}(t_0))(t - t_0) + \ddot{C}(\overline{x}(t_0))(t - t_0)^2 + O((t - t_0)^3). \quad (6\text{–}30)$$

Ignoring the $O((t - t_0)^3)$ term, we solve for $C(\overline{x}(t_c)) = 0$ to obtain the estimation

$$t_c = t_0 + \frac{-\dot{C}(\overline{x}(t_0)) \pm \sqrt{\dot{C}(\overline{x}(t_0))^2 - 2\ddot{C}(\overline{x}(t_0))C(\overline{x}(t_0))}}{\ddot{C}(\overline{x}(t_0))}. \quad (6\text{–}31)$$

[2] It is not always clear exactly which constraint this should be. For example, if a vertex from polyhedron A is inside polyhedron B, which face of B did it pass through to get there? In cases like this, the most likely non-penetration constraint should be chosen. For most situations, the correct non-penetration constraint is easy to determine. If several non-penetration constraints appear to have been violated, for example, if multiple vertices of A lie inside B, the non-penetration constraint that appears to have been violated the earliest should be chosen. For situations involving multiple constraint violations, an incorrect choice is not fatal—at worst, it will fail to yield a good estimate for the collision time.

The estimate t_c should be the smallest real root of Equation (6–31) greater than t_0; if no such root exists, or $\ddot{C}(\overline{x}(t_0))$ is near zero, t_c can be estimated using a linear approximation of $C(\overline{x}(t))$. The method is made robust by incorporating a bisection step when convergence is slow[15,44]; that is, predicting $t_c = (t_0 + t_1)/2$. For constant acceleration, Equation (6–31) gives an exact result for t_c, if $C(\overline{x}(t_0))$ is a linear measure of distance.

6.6 Rolling contact

As we have noted, $\ddot{C}(\overline{x}(t_0))$ is a measure of relative acceleration normal to the contact surfaces of A and B. In Chapter 9, when friction is discussed, we will need a description of both the relative slip velocity between A and B tangent to the contact surface, and the derivative of the slip velocity with respect to time. These quantities are expressed more naturally in world space than configuration space. The derivation of these quantities is straightforward for bodies of any geometry, and is certainly well known[4,41], but we will derive it here for completeness. We will also show why the constraint of pure rolling motion between bodies of arbitrary geometry is so much simpler to express than the non-penetration constraint we have just developed.

Let v_a, v_b, ω_a and ω_b denote the linear and angular velocities of A and B at time t_0. Let c_a and c_b denote the centers of mass of A and B. Then $\dot{c}_a = v_a$ and $\dot{c}_b = v_b$. Assume that A and B do not break contact near p_c. The slip velocity, s, between A and B at time t_0 is therefore

$$s = (v_a + \omega_a \times (p_c - c_a)) - (v_b + \omega_b \times (p_c - c_b)). \tag{6–32}$$

The derivative of the slip velocity is

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{dt}}s &= \frac{\mathrm{d}}{\mathrm{dt}}v_a + \frac{\mathrm{d}}{\mathrm{dt}}\omega_a \times (p_c - c_a) + \omega_a \times (\dot{p}_c - v_a) \\ &\quad - (\frac{\mathrm{d}}{\mathrm{dt}}v_b + \frac{\mathrm{d}}{\mathrm{dt}}\omega_b \times (p_c - c_b) + \omega_b \times (\dot{p}_c - v_b)). \qquad (6\text{–}33)\end{aligned}$$

Assuming that $s = \mathbf{0}$, that is, that A and B are not initially slipping, A and B can be made to maintain $s = \mathbf{0}$, and thus roll without slipping, by enforcing

$$\frac{\mathrm{d}}{\mathrm{dt}}s = \mathbf{0}. \qquad (6\text{–}34)$$

Since $\dot{p}_c$ is fairly easily calculated (by the previous section), the constraint that two bodies roll without slipping is easily derived as a constraint on the forces acting on A and B. Note that this constraint automatically enforces non-penetration between A and B. In order for A and B to inter-penetrate, they must attain some non-zero relative velocity normal to the contact surfaces; the constraint $\frac{\mathrm{d}}{\mathrm{dt}}s = \mathbf{0}$ prevents this from occurring. Why should this rolling without slipping constraint be so much simpler to express than simply the non-penetration constraint?

The motion of rolling without slipping between two bodies is a highly constrained motion. Equation (6–34) enforces three constraints on the relative acceleration, while Equation (6–4) enforces only one constraint. Thus, rolling without slipping has fewer degrees of freedom than simply moving without inter-penetrating. As a result, we have more knowledge about the motions of A and B, and the constraint is formulated much more easily.

6.7 Ill-conditioned Jacobians

As stated in Chapter 3, impact is assumed to have been dealt with prior to the computation of constraint forces. However, what has really been dealt with are situations where A and B have a non-zero approach velocity at p_c normal

to the contact surfaces. This is not the only case in which impact can arise. Featherstone[13] gives an example where discontinuities in the curvature of the contact surfaces require an impulse to be applied, even though the approach velocity normal to the contact surfaces is zero. Given the equations developed to compute $\ddot{C}$ for curved surfaces, we can add yet another type of impact that might occur between bodies.

Suppose we wish to simulate the frictionless oscillation of an ellipsoid A upon a plane surface B (Figure 6.6). Then the function F describing A is a simple quadratic function with non-zero curvature everywhere. B is described by a linear function G and has zero curvature everywhere. By solving Equation (6–25) for the reaction force between A and B, and considering Equation (6–29), it can be shown that for given initial conditions, the reaction force magnitude R is of the form

$$R(t) = O(\|\dot{p}_c(t)\| + 1) \tag{6–35}$$

over the simulation, treating p_c as a function of time.

Recall from Equation (6–22) that $\dot{p}_c$ is calculated as the ratio of two determinants. Let us calculate the denominator determinant J at a time t_0 when A and B are in contact (and thus $f(p_{ax}, p_{ay}, \overline{x}(t_0)) - g(p_{bx}, p_{by}, \overline{x}(t_0)) = 0$). Evaluating Equation (6–23), and since G (and thus g) have second partial derivatives everywhere,

$$J = \begin{vmatrix} f_{11} & f_{12} & 0 & 0 \\ f_{12} & f_{22} & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & -1 & 0 & -1 \end{vmatrix} = f_{11}f_{22} - f_{12}^2 = \kappa \tag{6–36}$$

where

$$f_{11} = \frac{\partial f}{\partial x^2}(p_{ax}, p_{ay}, \overline{x}(t_0)), \quad f_{22} = \frac{\partial f}{\partial y^2}(p_{ax}, p_{ay}, \overline{x}(t_0)), \tag{6–37}$$

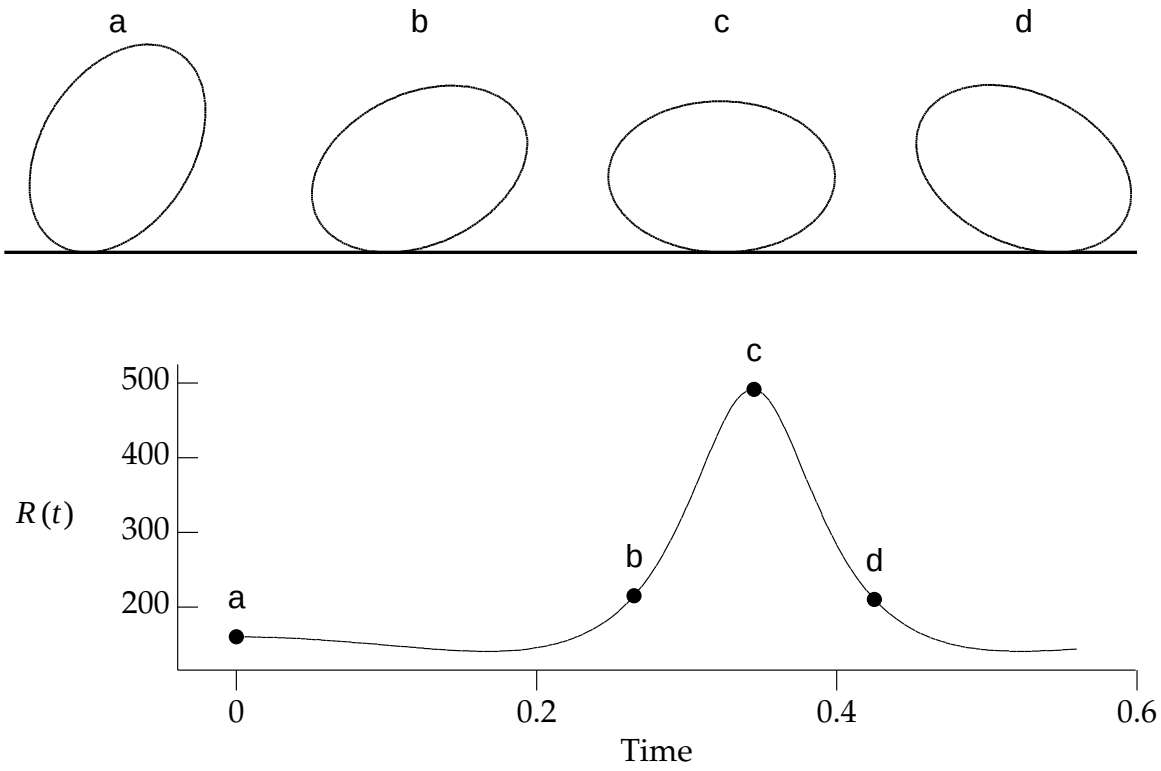

Figure 6.6: Frictionless oscillation of an ellipsoid A on a plane B.

and

$$f_{12} = \frac{\partial^2 f}{\partial x \partial y}(p_{ax}, p_{ay}, \overline{x}(t_0)). \tag{6–38}$$

This gives $J = \kappa$ where κ is the curvature of the ellipsoid at the contact point p_c. For the case of an ellipsoid, the curvature κ is always positive, so J is always non-zero. However, we can make κ arbitrarily close to zero by scaling the ellipsoid as shown in Figure 6.7. As the curvature at the contact point decreases, the reaction force increases, in a continuous fashion. The simulator must take small steps in order to accurately sample how the reaction force R changes over time. However, this is merely a practical problem of simulation. Mathematically speaking, nothing unusual occurs.

Suppose however that we enable A to have zero curvature at certain points. We can do this by letting A be a superquadric ellipsoid, defined in some rest frame as the set of points p satisfying

$$p_x^4 + p_y^4 + p_z^4 = 1. \tag{6–39}$$

Again, we will let A undergo frictionless oscillation in contact with B. Suppose that a point of zero curvature on A comes into contact with B at time t_0. As t approaches t_0, the value of J approaches zero, in a continuous fashion. As a result, $R(t)$ increases in a continuous fashion without bound, as t approaches t_0. In the limit, the reaction force $R(t)$ approaches infinity as the contact point approaches the zero curvature spot of A (Figure 6.8). Should this be considered as an impulsive reaction between A and B?

The answer to this is "almost". Clearly, the reaction force between A and B at t_0 cannot be considered to be a force, since it is infinite. However, in order for the reaction force to behave impulsively, it must cause a velocity discontinuity at time t_0. Even though the reaction force is unbounded, it does not cause such a discontinuity in velocity, although it does cause a discontinuity in acceleration.

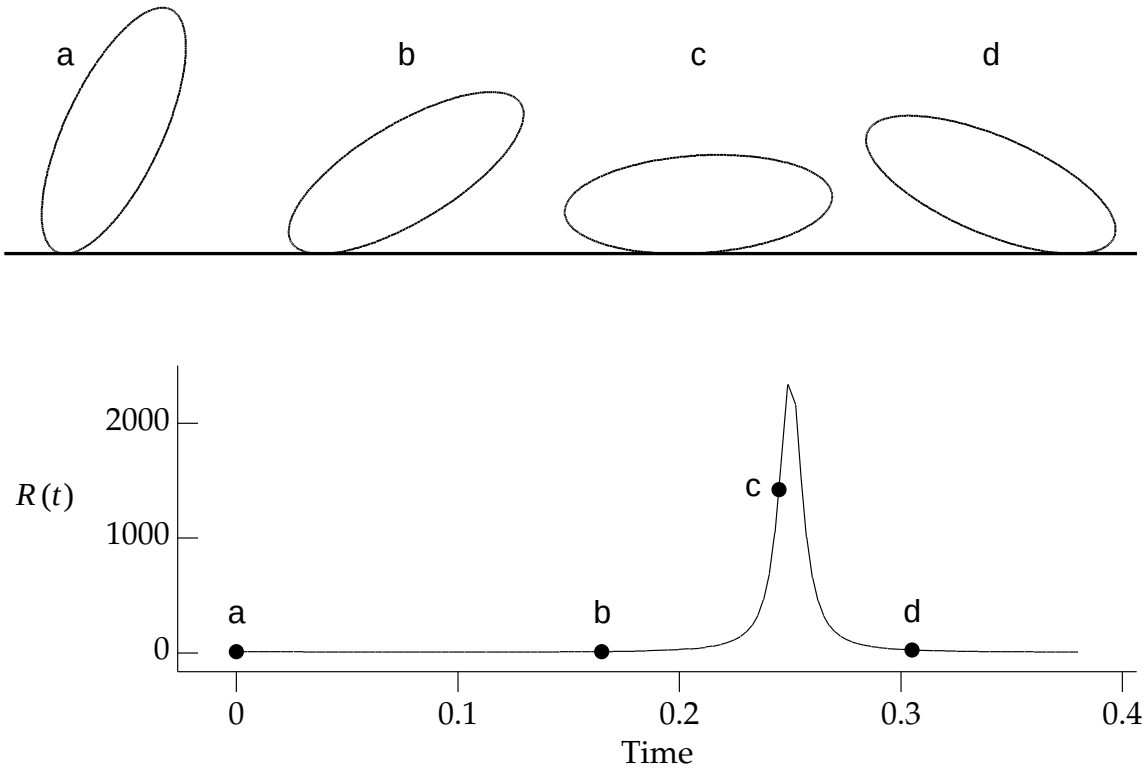

Figure 6.7: Frictionless oscillation of a more eccentric ellipsoid A on a plane B. The reaction force $R(t)$ is stronger and more sharply peaked the closer the curvature κ of A approaches zero.

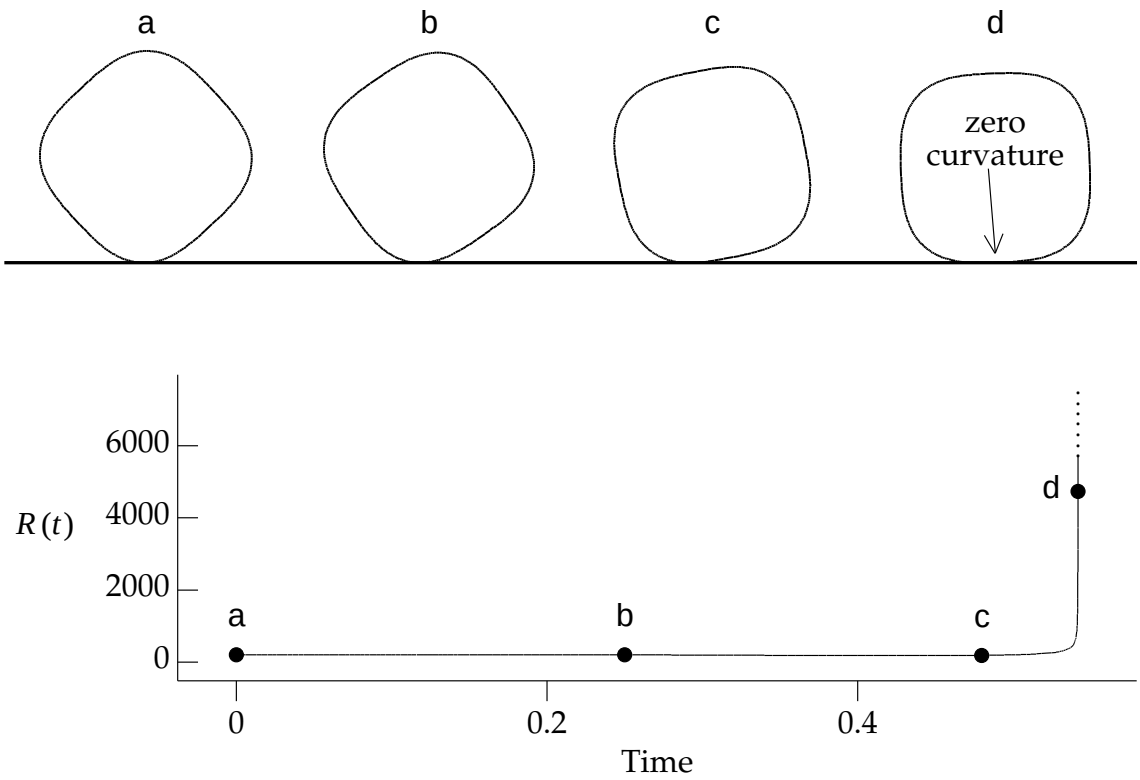

Figure 6.8: Frictionless oscillation of a superquadric ellipsoid A on a plane B. The reaction force $R(t)$ diverges as the contact point approaches a zero curvature spot on the superquadric ellipsoid A.

That is, the reaction force at time t_0 is an infinite force acting for an instant of time, but in such a way that the measured impulse is zero. To see this, consider the integral of the reaction force R over time. In order for the reaction force to behave as an impulse, it is necessary that

$$\lim_{\Delta t \to 0} \int_{t_0 - \Delta t}^{t_0 + \Delta t} R(t)\, dt > 0. \tag{6–40}$$

Since $R(t) = O(\|\dot{p}_c(t)\| + 1)$, consider

$$\lim_{\Delta t \to 0} \int_{t_0 - \Delta t}^{t_0 + \Delta t} O(\|\dot{p}_c(t)\| + 1)\, dt.$$

Since this is an integral over time involving $\dot{p}_c(t)$, by the fundamental theorem of calculus there exists a continuous function H of p_c such that

$$\int_{t_0 - \Delta t}^{t_0 + \Delta t} O(\|\dot{p}_c(t)\| + 1)\, dt = H(p_c(t_0 + \Delta t)) - H(p_c(t_0 - \Delta t)). \tag{6–41}$$

Since $p_c(t)$ is a spatial measure (the position of the contact point at time t), $p_c(t)$ is clearly continuous. Therefore,

$$\begin{aligned} \lim_{\Delta t \to 0} \int_{t_0 - \Delta t}^{t_0 + \Delta t} R(t)\, dt &= \lim_{\Delta t \to 0} \int_{t_0 - \Delta t}^{t_0 + \Delta t} O(\|\dot{p}_c(t)\| + 1)\, dt \\ &= \lim_{\Delta t \to 0} H(p_c(t + \Delta t)) - H(p_c(t - \Delta t)) \qquad (6–42) \\ &= H(p_c(t_0)) - H(p_c(t_0)) \\ &= 0. \end{aligned}$$

Thus, the reaction force is not impulsive. Viewed in another way, the reaction force behaves in the limit as an impulse of zero magnitude. For want of a better term, we could call such behavior a "zeroeth order" impact.

In practical terms, the problem of a zeroeth order impact is how to evaluate the integral of the force over a time period containing the impact. Current ordinary

differential equation solvers are not well suited for evaluating this integral, even though it is finite. From a sampling viewpoint, the reaction force must be sampled infinitely often near time t_0 to accurately evaluate the integral of the reaction force. One approach is to analytically evaluate the integral in a region of time bounding t_0, but it is unknown how to do this efficiently in a simulation environment. The simulation process described in this thesis does not deal with zeroeth order impacts.

Chapter 7

The Penalty Method

The remainder of this thesis will be devoted to the problem of computing the forces that arise at contact points for systems with multiple contact points. In this chapter, we will examine the penalty method for computing constraint forces between bodies, in the absence of friction. Since we already know that the penalty method usually performs poorly in practice, why bother to consider it at all? The main reason for considering this method is that in Chapter 12, we shall consider a model of frictional behavior that is based on the penalty method. Prior to this, we need to consider whether the penalty method for frictionless systems yields physically correct motion. Validating the penalty method for the frictionless case does not necessarily validate the model of frictional behavior proposed in Chapter 12, but it would clearly be unreasonable to extrapolate from the penalty method if it were known to be flawed for the case of frictionless systems. A secondary reason for considering the penalty method is to gain insights into its correctness as a simulation method; considering the large amount of simulation work that uses the penalty method, this is a topic worth pursuing.

7.1 Equality constrained motion

To simplify matters, we will first consider the penalty method as applied to equality constrained dynamics problems. This is actually not a large restriction, because systems with inequality constraints can be viewed as being subject to the corresponding equality constraints on any time interval during which no constraint changes from active to inactive or vice versa. For example, in the particle/surface example of Section 2.5, the motion of the particle, whenever it is in contact with the surface, is exactly the same as if it were constrained to always lie on the surface. Thus, we would first like to study the motion produced by the penalty method for equality constrained problems, such as the problem described in Section 2.4.

Rubin and Ungar[45] give an excellent proof that the penalty method does indeed yield (in the limit) physically correct behavior for frictionless systems with equality constraints. However, their proof assumes that the external forces acting on bodies are due to the potential energy of the system, rather than being arbitrary functions of time. Modifying Rubin and Ungar's proof to handle arbitrary time-dependent external forces does not seem overly difficult, but their proof is fairly lengthy and somewhat complicated, and proves a much stronger result than we are interested in. We will show that the penalty method produces (in the limit) physically correct motion by comparing differential descriptions of the physically correct motion and the motion produced by the penalty method. This is much simpler and more intuitive than Rubin and Ungar's approach; note however that Rubin and Ungar obtain a much stronger result about the limiting motion of the penalty method than can be achieved using the simple differential comparison.

Let us consider a system with equality constraints from time t_0 to t_1. In the previous chapter, the non-penetration constraint at a single contact point was represented by a hypersurface in C-space, and the configuration $\overline{x}(t)$ was required to

lie on or "above" this hypersurface. In restricting ourselves to equality constraints, we are requiring $\overline{x}(t)$ to lie on the hypersurface. For a system with k bodies and n contact points, the n non-penetration constraints are represented as n hypersurfaces in a $6k$-dimensional C-space. The configuration point $\overline{x}(t)$ must lie on all n hypersurfaces. Equivalently, $\overline{x}(t)$ must lie in the *intersection* of all the constraint hypersurfaces. Since the constraint hypersurfaces are manifolds, the intersection of the constraint hypersurfaces is itself a manifold, which can be expressed in terms of a scalar function $C_I(\overline{p})$ as $C_I(\overline{p}) = 0$. We shall call this manifold the *intersection manifold*.

7.1.1 The physically correct motion

We will use Lagrangian dynamics to derive the equation of motion for a system constrained to satisfy $C_I(\overline{x}(t)) = 0$ from time t_0 to t_1. The remainder of this chapter assumes a knowledge of Lagrangian dynamics and differentiable manifolds.

In order to describe the motion of the system (that is, the function $\overline{x}(t)$ from t_0 to t_1), we will have to describe the configuration state $\overline{x}(t)$, and velocity $\dot{\overline{x}}(t)$, in terms of a coordinate system on the intersection manifold. Let the intersection manifold have dimension d, and let $S(q)$ be a parameterization of the manifold. The function $S(q)$ invertibly maps points from some region of $\mathbf{R}^d$ to the intersection manifold. (If the manifold cannot be covered with a single parameterization, the motion of $\overline{x}(t)$ can be considered in suitably small intervals.) The motion of $\overline{x}(t)$, restricted to lie on the intersection manifold is then described in terms of a continuous path $q(t)$ in $\mathbf{R}^d$. Given initial values for $\overline{x}(t_0)$ and $\dot{\overline{x}}(t_0)$, the initial conditions $q(t_0)$ and $\dot{q}(t_0)$ must satisfy

$$S(q(t_0)) = \overline{x}(t_0) \qquad \text{and} \qquad \frac{\mathrm{d}}{\mathrm{dt}} S(q(t_0)) = dS_{q(t_0)}\dot{q}(t_0) = \dot{\overline{x}}(t_0) \qquad (7\text{–}1)$$

where

$$dS_{q(t_0)} = \frac{\partial S}{\partial q}(q(t_0)). \tag{7–2}$$

The function $dS_{q(t)}$ maps vectors from $\mathbf{R}^d$ to vectors that are tangent to the intersection manifold at $S(q(t))$.

We also need to describe the kinetic energy of the system. Given a system with configuration $\overline{p}$ in C-space and velocity $\overline{v}$, the kinetic energy T of the system can be written in the form

$$T = \tfrac{1}{2}\overline{v}^T M(\overline{p})\overline{v} \tag{7–3}$$

where $M(\overline{p})$ is a square matrix with size equal to the dimension of C-space. The matrix $M(\overline{p})$ is symmetric positive definite. The kinetic energy is used to induce an inner product on C-space; given vectors $\overline{u}$ and $\overline{w}$ at point $\overline{p}$ of C-space, the inner product of $\overline{u}$ and $\overline{w}$ is written $<\overline{u},\overline{w}>_{\overline{p}}$ and is defined as

$$<\overline{u},\overline{w}>_{\overline{p}} = <\overline{w},\overline{u}>_{\overline{p}} = \overline{u}^T M(\overline{p})\overline{w}. \tag{7–4}$$

Using the inner product, T can be written as

$$T = \tfrac{1}{2}<\overline{v},\overline{v}>_{\overline{p}}. \tag{7–5}$$

Expressing the kinetic energy T in terms of the parametric coordinates of $\overline{x}(t)$, we have

$$\begin{aligned} T &= \tfrac{1}{2}<\dot{\overline{x}}(t),\dot{\overline{x}}(t)>_{\overline{x}(t)} \\ &= \tfrac{1}{2}<\frac{\mathrm{d}}{\mathrm{dt}}S(q(t)),\frac{\mathrm{d}}{\mathrm{dt}}S(q(t))>_{S(q(t))} \\ &= \tfrac{1}{2}<dS_{q(t)}\dot{q}(t),dS_{q(t)}\dot{q}(t)>_{S(q(t))}. \end{aligned} \tag{7–6}$$

(Henceforth, all evaluations of dS will take place at $q(t)$, so we shall simply write dS in place of $dS_{q(t)}$. Likewise, all inner products are assumed to be evaluated at $S(q(t))$ unless otherwise qualified.) Given any external C-space force $F(t)$ acting on

the system, let $Q(t)$ be the equivalent generalized force on $q(t)$. If we assume that the potential energy of the system is always constant, since $Q(t)$ can include the effects of forces due to potentials, the equation of motion for $q(t)$ is given by

$$\frac{\mathrm{d}}{\mathrm{dt}}\left(\frac{\partial T}{\partial \dot{q}}\right) - \frac{\partial T}{\partial q} = Q. \tag{7–7}$$

Using Equation (7–6), this yields

$$\frac{\mathrm{d}}{\mathrm{dt}}\left(\frac{\partial}{\partial \dot{q}}\left(\tfrac{1}{2}<dS\dot{q}(t), dS\dot{q}(t)>\right)\right) - \frac{\partial}{\partial q}\left(\tfrac{1}{2}<dS\dot{q}(t), dS\dot{q}(t)>\right) = Q(t) \tag{7–8}$$

with initial conditions for $q(t_0)$ and $\dot{q}(t_0)$ as specified by Equation (7–1).

7.1.2 Motion due to the penalty method

Next, we will derive an equation of motion for the penalty method, which will then be compared with Equation (7–8). As defined in Section 2.4, the penalty method works by imposing a penalty force on the system when $\overline{x}(t)$ deviates from the intersection manifold. The penalty force can be viewed as the gradient of a potential energy function $P(\overline{p})$ that increases the farther $\overline{p}$ lies from the intersection manifold. Thus, the penalty force is conservative, and does not change the overall energy of the system. If the potential energy function P is made steep enough, then for a conservative system, $\overline{x}(t)$ can never deviate more than a certain distance from the intersection manifold; that distance is determined by the total energy of the system. Thus, for the case when the only external forces acting on the system are due to the potential energy of the system, increasing the penalty force (by increasing the steepness of P) binds $\overline{x}(t)$ closer to the intersection manifold.

However, if time-dependent, and thus, non-conservative, external forces act on the system, increasing the slope of P does not automatically limit the deviation of $\overline{x}(t)$ from the intersection manifold. If the external force varies over time, it is possible that a steeper P may cause a "resonance", and allow $\overline{x}(t)$ to deviate more

from the intersection manifold. For example, the amount of work done by a given external driving force on a linear spring system is not monotonic with respect to the spring constant. In general though, as the spring constant is increased, the driving force does less work. Similarly, in analyzing the penalty method, we will assume that the maximum deviation of $\overline{x}(t)$ from the intersection manifold can be made arbitrarily small, by increasing the steepness of P in such a way that resonance is avoided. We would like to show that as the maximum deviation decreases, the path traced in the vicinity of the intersection manifold converges to the physically correct path lying exactly on the intersection manifold.

In the previous section, the position of $\overline{x}(t)$ on the intersection manifold was described by the parametric coordinate $q(t)$. For the penalty method, we need to describe $\overline{x}(t)$ in terms of a coordinate $q(t)$ on the intersection manifold, and a displacement from the intersection manifold. To accomplish this, we will describe a coordinate system for the orthogonal complement of the intersection manifold. Let the orthogonal complement of the intersection manifold have dimension k. At each point $\overline{p}$ on the manifold, we define vectors $N_1(\overline{p})$ through $N_k(\overline{p})$ such that

$$<N_i(\overline{p}), N_i(\overline{p})>_{\overline{p}} = 1 \qquad (1 \leq i \leq k) \tag{7–9}$$

and

$$<N_i(\overline{p}), N_j(\overline{p})>_{\overline{p}} = 0 \qquad (1 \leq i < j \leq k). \tag{7–10}$$

If $\overline{u}$ is tangent to the intersection manifold at $\overline{p}$, then

$$<N_i(\overline{p}), \overline{u}>_{\overline{p}} = 0 \qquad (1 \leq i \leq k); \tag{7–11}$$

that is, each $N_i(\overline{p})$ is normal to the intersection manifold at $\overline{p}$, as defined by the inner product induced by the kinetic energy. Each $N_i(\overline{p})$ varies smoothly over the intersection manifold; since we are considering the motion of $\overline{x}(t)$ locally, orientability of the intersection manifold is not a concern.

Given this orthogonal coordinate system, we will express points $\overline{x}(t)$ lying near the intersection manifold using the parametric coordinate $q(t)$ and a k-tuple of real numbers $(h_1(t), \ldots, h_n(t))$ as

$$\overline{x}(t) = S(q(t)) + \sum_{i=1}^{k} h_i(t) N_i(S(q(t))). \tag{7–12}$$

This coordinate system is one-to-one in some region of C-space properly containing the intersection manifold, as long as the curvature of the manifold is bounded. Since we can force $\overline{x}(t)$ to lie as close to the intersection manifold as necessary, Equation (7–12) is an acceptable way of defining the position of $\overline{x}(t)$.

Using the notation of Equation (7–12), it is simple to write an equation of motion for the penalty method. Since the values $h_i(t)$ describe the displacement of $\overline{x}(t)$ from the constraint manifold, we can write the potential energy of the penalty function in terms of $h_i(t)$. The assumption is that for any $\epsilon > 0$, a potential energy function $P(h_1(t), \ldots, h_k(t))$ can be chosen[1] so that $|h_i(t)| < \epsilon$ for all $1 \leq i \leq k$ and $t_0 \leq t \leq t_1$.

The kinetic energy T' of the system, as in Equation (7–6), is given by

$$T' = \tfrac{1}{2} < \dot{\overline{x}}(t), \dot{\overline{x}}(t) >. \tag{7–13}$$

From Equation (7–12),

$$\dot{\overline{x}}(t) = dS\dot{q}(t) + \sum_{i=1}^{k} \dot{h}_i(t) N_i(S(q(t))) + \sum_{i=1}^{k} h_i(t) \dot{N}_i(S(q(t))) \tag{7–14}$$

where $\dot{N}_i(S(q(t))) = \frac{\mathrm{d}}{\mathrm{dt}} N_i(S(q(t)))$. Then the kinetic energy can be written as

[1] This does not mean to imply that such a choice is simple—it's not. The choice of a suitably strong penalty forces usually requires great care. This is simply one of the penalty method's many problems.

$$
\begin{aligned}
T' &= \tfrac{1}{2}<\dot{\bar{x}}(t), \dot{\bar{x}}(t)> \\
&= \tfrac{1}{2}<dS\dot{q}(t) + \left(\sum_{i=1}^{k} \dot{h}_i(t) N_i(S(q(t)))\right) + \left(\sum_{i=1}^{k} h_i(t) \dot{N}_i(S(q(t)))\right), \\
&\qquad dS\dot{q}(t) + \left(\sum_{i=1}^{k} \dot{h}_i(t) N_i(S(q(t)))\right) + \left(\sum_{i=1}^{k} h_i(t) \dot{N}_i(S(q(t)))\right)> \\
&= \tfrac{1}{2}\Big(<dS\dot{q}(t), dS\dot{q}(t)> \qquad (7\text{–}15) \\
&\qquad + <\sum_{i=1}^{k} \dot{h}_i(t) N_i(S(q(t))), \sum_{i=1}^{k} \dot{h}_i(t) N_i(S(q(t)))> \\
&\qquad + <\sum_{i=1}^{k} h_i(t) \dot{N}_i(S(q(t))), \sum_{i=1}^{k} h_i(t) \dot{N}_i(S(q(t)))>\Big) \\
&\quad + <dS\dot{q}(t), \sum_{i=1}^{k} \dot{h}_i(t) N_i(S(q(t)))> + <dS\dot{q}(t), \sum_{i=1}^{k} h_i(t) \dot{N}_i(S(q(t)))> \\
&\quad + <\sum_{i=1}^{k} \dot{h}_i(t) N_i(S(q(t))), \sum_{i=1}^{k} h_i(t) \dot{N}_i(S(q(t)))>.
\end{aligned}
$$

We can simplify this expression for T' by noting that the vectors $N_1(S(q(t)))$ through $N_k(S(q(t)))$ are orthonormal with respect to the inner product. Thus

$$
<\sum_{i=1}^{k} \dot{h}_i(t) N_i(S(q(t))), \sum_{i=1}^{k} \dot{h}_i(t) N_i(S(q(t)))> = \sum_{i=1}^{k} \dot{h}_i(t)^2. \qquad (7\text{–}16)
$$

Also, since the vector $dS\dot{q}(t)$ is tangent to the intersection manifold at $q(t)$ for any $\dot{q}(t)$, the inner product $<dS\dot{q}(t), N_i(S(q(t)))>$ is also zero for any $1 \leq i \leq k$ and any $\dot{q}(t)$. Thus,

$$
<dS\dot{q}(t), \sum_{i=1}^{k} \dot{h}_i(t) N_i(S(q(t)))> = 0. \qquad (7\text{–}17)
$$

Using these facts, we may rewrite Equation (7–15) as

$$
\begin{aligned}
T' &= \tfrac{1}{2}\Big(<dS\dot{q}(t), dS\dot{q}(t)> + \sum_{i=1}^{k} \dot{h}_i(t)^2 \\
&\qquad + <\sum_{i=1}^{k} h_i(t) \dot{N}_i(S(q(t))), \sum_{i=1}^{k} h_i(t) \dot{N}_i(S(q(t)))>\Big) \qquad (7\text{–}18) \\
&\quad + <dS\dot{q}(t), \sum_{i=1}^{k} h_i(t) \dot{N}_i(S(q(t)))>
\end{aligned}
$$

$$+ <\sum_{i=1}^{k} \dot{h}_i(t) N_i(S(q(t))), \sum_{i=1}^{k} h_i(t) \dot{N}_i(S(q(t)))>.$$

The equation of motion for $q(t)$ is

$$\frac{\mathrm{d}}{\mathrm{dt}} \left(\frac{\partial}{\partial \dot{q}} (T' - P) \right) - \frac{\partial}{\partial q} (T' - P) = Q, \tag{7–19}$$

with the same initial conditions of the physically correct motion specified by Equation (7–1). Since the penalty energy function P is independent of both $q(t)$ and $\dot{q}(t)$, we can express Equation (7–19) as

$$\frac{\mathrm{d}}{\mathrm{dt}} \left(\frac{\partial T'}{\partial \dot{q}} \right) - \frac{\partial T'}{\partial q} = Q. \tag{7–20}$$

This equation is independent of the penalty force $-\nabla P$; as expected, the penalty force imparts no "tangential" acceleration to $\overline{x}(t)$, along the intersection manifold.

All that remains is to show that the motion of $q(t)$ according to Equation (7–19) is the same as the motion according to Equation (7–8) in the limit as $h_i(t) \to 0$ for each $h_i(t)$. Since the physically correct motion and the penalty method motion both have the same initial conditions, all we need to do is show that as the $h_i(t)$s converges to zero, Equation (7–19) and Equation (7–8) yield the same differential equation.

First, consider $\partial T'/\partial \dot{q}$. Since we are not differentiating with respect to h_i, any terms containing $h_i(t)$ drop out as $h_i \to 0$. Differentiating Equation (7–18) with respect to $\dot{q}$, all that remains is

$$\begin{aligned} \frac{\partial T'}{\partial \dot{q}} &= \frac{\partial}{\partial \dot{q}} \left(\tfrac{1}{2} <dS\dot{q}(t), dS\dot{q}(t)> + \sum_{i=1}^{k} \dot{h}_i(t)^2 \right) \\ &= \frac{\partial}{\partial \dot{q}} \left(\tfrac{1}{2} <dS\dot{q}(t), dS\dot{q}(t)> \right). \end{aligned} \tag{7–21}$$

Now, consider $\partial T'/\partial q$. Again, any terms containing $h_i(t)$ drop out, and we have

$$\frac{\partial T'}{\partial q} = \frac{\partial}{\partial q} \left(\tfrac{1}{2} <dS\dot{q}(t), dS\dot{q}(t)> + \sum_{i=1}^{k} \dot{h}_i(t)^2 \right)$$

$$= \frac{\partial}{\partial q}\left(\tfrac{1}{2}<dS\dot{q}(t), dS\dot{q}(t)>\right). \qquad (7\text{–}22)$$

Thus, as the $h_i(t)$s converge to zero, the equation of motion

$$\frac{\mathrm{d}}{\mathrm{dt}}\left(\frac{\partial T'}{\partial \dot{q}}\right) - \frac{\partial T'}{\partial q} = Q \qquad (7\text{–}23)$$

becomes

$$\frac{\mathrm{d}}{\mathrm{dt}}\left(\frac{\partial}{\partial \dot{q}}\left(\tfrac{1}{2}<dS\dot{q}(t), dS\dot{q}(t)>\right)\right) - \frac{\partial}{\partial q}\left(\tfrac{1}{2}<dS\dot{q}(t), dS\dot{q}(t)>\right) = Q(t). \qquad (7\text{–}24)$$

Since this is precisely the same as Equation (7–8), the motion due to the penalty method converges to the physically correct motion as the maximum deviation of $\overline{x}(t)$ from the intersection manifold converges to zero. The proof by Rubin and Ungar establishes that the normal velocity, which is described by $\dot{h}_1(t)$ through $\dot{h}_k(t)$ in our parameterization, also converges to zero as the steepness of the penalty function P is increased.

7.2 Inequality constrained motion

The penalty method yields a correct result (in the limit) on time intervals during which no constraints change. However, what happens at the instant that a constraint *does* change? Specifically, how does the penalty method know when to "let go" of a contact point, and allow the contact to break? (The converse of this involves a constraint activating, which results in an impulse, but we are assuming that impulses have already been dealt with.) For non-penetration constraints, the penalty force for a given non-penetration constraint is set to zero if the constraint has not been violated; this prevents penalty forces from becoming attractive. Unfortunately, this does not always give the correct effect, because in some situations the penalty force must act attractively to correctly maintain contact. Thus, one cannot simply release

a constraint when the penalty force would have become attractive. It would be nice to speculate that the normal *accelerations*, that is, $\ddot{h}_1(t)$ through $\ddot{h}_k(t)$, converge to zero as the deviation from the intersection manifold goes to zero. If that were so, then it is possible that, given a very small step size Δt, examination of $\ddot{h}_1(t)$ through $\ddot{h}_k(t)$ would indicate whether a constraint should be made inactive; presumably, an increasing value of $\ddot{h}_i(t)$ would indicate that the ith contact point should be released. Unfortunately, Rubin and Ungar give an example that shows that in general $\ddot{h}_1(t)$ through $\ddot{h}_k(t)$ do not converge to zero.[2]

Thus, the penalty method must be used in conjunction with an algorithm that decides when constraints should be released. The algorithms in the next few chapters furnish precisely this information; but since they also calculate the constraint forces as well, they are clearly a better simulation method than the penalty method. The frictional model extrapolated from the penalty method in Chapter 12 is based on the behavior of the penalty method over an interval in which constraints do not change state.

[2] Actually, they don't converge to anything.

Chapter 8

Frictionless Systems

As we saw in the previous chapter, the problem of simulating non-penetration constraints on intervals when there is no change in constraint status is an equality constrained problem. Constraint forces for equality constrained problems can be computed by solving a system of linear equations. However, for the general problem of computing the constraint forces at time t_0 for a system with n contact points, we must allow for the possibility that some set of the constraints may become inactive after time t_0. The inability of the penalty method to determine which constraints should become inactive at time t_0 is a major shortcoming of the method.

Let us consider a system of n contact points without friction. At the ith contact point, the unit surface normal between bodies is denoted $\hat{n}_i$. If this contact occurs between bodies A and B, and $\hat{n}_i$ points outwards from B (towards A) as in Figure 6.4, the constraint force acting on A can be written as $f_{N\,i}\hat{n}_i$ where $f_{N\,i} \geq 0$. An equal and opposite force $-f_{N\,i}\hat{n}_i$ acts on B. The magnitude of the ith constraint force is thus $f_{N\,i}$. The n-vector collection of constraint force magnitudes is denoted $\boldsymbol{f_N}$; the ith element of $\boldsymbol{f_N}$ is $f_{N\,i}$. (In general, matrices and vectors will be denoted by boldface type; for example, "$\boldsymbol{f_N}$". Components of matrices and vectors are

written in regular type; for example, "f_{Ni}".)

If a constraint function $C_i(\overline{x}(t))$ is formulated for the ith contact point, then $\ddot{C}_i(\overline{x}(t_0))$ is a measure of the relative normal acceleration at the contact point, as described in Chapter 6. For simplicity, we will denote this normal acceleration $\ddot{C}_i(\overline{x}(t_0))$ by the quantity a_{Ni}. The n-vector collection of normal accelerations is denoted $\mathbf{a_N}$. The vector $\mathbf{a_N}$ can be considered a function of the unknown constraint force magnitudes $\boldsymbol{f_N}$. Since acceleration is a linear function of force, $\mathbf{a_N}$ can be expressed, using the derivations of Chapter 6, as

$$\mathbf{a_N} = \mathbf{A}\boldsymbol{f_N} - \mathbf{b} \tag{8–1}$$

for some $n \times n$ matrix $\mathbf{A}$ and n-vector $\mathbf{b}$.

The matrix $\mathbf{A}$ reflects both the inverse inertias of the contact bodies, and the contact geometry. The vector $\mathbf{b}$ reflects relative normal accelerations at contact points due to known forces; this includes both inertial effects and acceleration due to known external forces. The matrix $\mathbf{A}$ is well known to be positive semidefinite (PSD) and symmetric. The vector $\mathbf{b}$ is always in the range of $\mathbf{A}$, that is,

$$\mathbf{b} = \mathbf{A}\mathbf{c} \tag{8–2}$$

for some n-vector $\mathbf{c}$; situations in which this is not so represent unsatisfiable kinematic constraints on the system and are not of concern to us.

Since inter-penetration is prevented by requiring the relative normal accelerations at each contact point to be non-negative, the non-penetration constraint for the entire system can be written as

$$a_{Ni} \geq 0 \qquad (1 \leq i \leq n). \tag{8–3}$$

We denote this as $\mathbf{a_N} \geq \mathbf{0}$ or, using Equation (8–1), as

$$\mathbf{A}\boldsymbol{f_N} - \mathbf{b} \geq \mathbf{0}. \tag{8–4}$$

Likewise, the constraint that each $f_{N_i} \geq 0$ is written as

$$\boldsymbol{f_N} \geq \mathbf{0}. \tag{8–5}$$

Recall from Section 2.5.2 that a constraint force, in addition to being non-negative, also has to be zero when the constraint is being broken. As in Section 2.5.2, this can be expressed as the constraint

$$(f_{N_i})(a_{N_i}) = 0 \qquad (1 \leq i \leq n). \tag{8–6}$$

Since $\boldsymbol{f_N} \geq \mathbf{0}$ and $\mathbf{a_N} = \mathbf{A}\boldsymbol{f_N} - \mathbf{b} \geq \mathbf{0}$ imply $\boldsymbol{f_N}^T\mathbf{a_N} \geq 0$, Equations (8–4) through (8–6) can be written as

$$\mathbf{a_N} = \mathbf{A}\boldsymbol{f_N} - \mathbf{b} \geq \mathbf{0}, \quad \boldsymbol{f_N} \geq \mathbf{0}, \qquad \text{and} \qquad \boldsymbol{f_N}^T\mathbf{a_N} = 0. \tag{8–7}$$

Equation (8–7) defines what is known as a *linear complementarity problem* (LCP); a vector $\boldsymbol{f_N}$ satisfying Equation (8–7) is a solution to the LCP. Thus, the constraint force magnitudes for a frictionless configuration can be computed by solving an LCP. We will call a set of normal forces satisfying Equation (8–7) a *valid* set of constraint forces.

LCP's are closely related to *quadratic programming*. A quadratic program (QP) has the general form

$$\underset{\mathbf{x}}{\text{minimize}} \ \mathbf{x}^T\mathbf{M}\mathbf{x} + \mathbf{c}^T\mathbf{x} \quad \text{subject to} \quad \left\{ \begin{array}{c} \mathbf{M}\mathbf{x} \geq \mathbf{d} \\ \mathbf{x} \geq \mathbf{0} \end{array} \right\}. \tag{8–8}$$

Any LCP can be written as a QP. Using

$$\boldsymbol{f_N}^T\mathbf{a_N} = \boldsymbol{f_N}^T(\mathbf{A}\boldsymbol{f_N} - \mathbf{b}) = \boldsymbol{f_N}^T\mathbf{A}\boldsymbol{f_N} - \boldsymbol{f_N}^T\mathbf{b}, \tag{8–9}$$

Equation (8–7), phrased as a QP, is

$$\underset{\boldsymbol{f_N}}{\text{minimize}} \ \boldsymbol{f_N}^T\mathbf{A}\boldsymbol{f_N} - \mathbf{b}^T\boldsymbol{f_N} \quad \text{subject to} \quad \left\{ \begin{array}{c} \mathbf{A}\boldsymbol{f_N} \geq \mathbf{b} \\ \boldsymbol{f_N} \geq \mathbf{0} \end{array} \right\}. \tag{8–10}$$

The objective function $\boldsymbol{f_N}^T\mathbf{A}\boldsymbol{f_N} - \boldsymbol{f_N}^T\mathbf{b}$ achieves zero for every $\boldsymbol{f_N}$ that satisfies Equation (8–7), and no others. Additionally, the QP formulation can directly represent equality constraints. If the ith constraint of the system is an equality constraint, the condition $f_{Ni} \geq 0$ in Equation (8–10) can be removed, and the condition $a_{Ni} \geq 0$ can be changed to $a_{Ni} = 0$. The QP in this case is still of size n. The same equality constraint can be indirectly represented in Equation (8–7) by a reformulation that adds the constraint $-a_{Ni} \geq 0$ and replaces the variable f_{Ni} with $f_{Ni} = f'_{Ni} - f''_{Ni}$ where $f'_{Ni} \geq 0$ and $f''_{Ni} \geq 0$. The resulting LCP is of size $n+1$ and is positive semidefinite, but not positive definite.

In the general case, determining if an LCP has a solution or a QP can achieve a certain minimum value are *NP*-complete problems[40, 52]. The corresponding problems of finding a solution to an LCP or finding a minimizer of a QP are *NP*-hard. However, when the coefficient matrices of the QP or LCP are PSD, then the problems have polynomial time solutions[40]. LCP's and QP's with PSD coefficient matrices are known as *convex* problems. In practice, convex QP's and LCP's are solved by efficient algorithms whose worst case behavior is exponential but whose expected running time is polynomial. We can use such an algorithm to compute $\boldsymbol{f_N}$ and determine the constraint forces for configurations without friction.

8.1 Existence and uniqueness of solutions

There are two basic questions about $\boldsymbol{f_N}$ that should be asked. First, does an $\boldsymbol{f_N}$ satisfying Equation (8–7) always exist? Furthermore, if an $\boldsymbol{f_N}$ exists, is it unique? (Equivalently: do valid constraint forces exist for every frictionless configuration? Are they unique?) For the case when the LCP of Equation (8–7) is convex, and $\mathbf{b}$ lies in the column-space of $\mathbf{A}$, an $\boldsymbol{f_N}$ satisfying the LCP always exists. If $\mathbf{A}$, in addition to being PSD, is non-singular, then $\mathbf{A}$ is positive definite (PD), and

$\boldsymbol{f_N}$ is unique. Otherwise, although $\boldsymbol{f_N}$ is not necessarily unique, $\mathbf{a_N} = \mathbf{A}\boldsymbol{f_N} - \mathbf{b}$ is unique; Cottle[7] gives an elegant proof of this fact. The non-uniqueness of $\boldsymbol{f_N}$ arises in cases where the distribution of constraint forces among contact points is underdetermined. The canonical example is of a chair resting on the floor on four or more legs. Although the acceleration of the chair is unique, the distribution of force on the legs is not.

8.2 Implementation issues

For frictionless configurations, the problem of determining a valid set of constraint forces is well in hand. We will conclude this chapter by discussing two implementation issues in computing $\boldsymbol{f_N}$.

The first issue involves avoiding computational effort by reusing information from previous time steps. It is well known in the mathematical programming literature that when Equation (8–7) has a solution, it has at least one solution called a *basic* solution. A basic solution may be quantified by a subset $L \subset \{1, 2, \ldots, n\}$ as follows. Let $\mathbf{B}$ denote the matrix obtained from $\mathbf{A}$ by deleting each row and column whose index is not in L; this yields a square sub-matrix of $\mathbf{A}$. Similarly, let $\mathbf{d}$ denote the vector obtained from $\mathbf{b}$ by deleting elements of $\mathbf{b}$ whose index is not in L.

The special properties of a basic solution are as follows. First, the matrix $\mathbf{B}$ is invertible. Second, the vector $\mathbf{B}^{-1}\mathbf{d}$ satisfies $\mathbf{B}^{-1}\mathbf{d} \geq \mathbf{0}$. Last, let us expand the vector $\mathbf{B}^{-1}\mathbf{d}$ by inserting zero elements. The zeroes are inserted into $\mathbf{B}^{-1}\mathbf{d}$ by inverting the process of deleting elements from $\mathbf{b}$ to form $\mathbf{d}$; the expanded vector has a zero for every index i not contained in L. The newly expanded vector is then a solution to Equation (8–7).

The concept of a basic solution also applies to the QP of Equation (8–10); a solution to Equation (8–10) can be obtained by inverting a matrix formed from the

coefficient matrices defining the QP. Most algorithms for solving LCP's and QP's find a basic solution. In a simulation environment, the concept of a basic solution can greatly reduce the amount of numerical work done at each time step. The major effort of computing the constraint forces lies in solving for $\boldsymbol{f_N}$. However, given a basic solution for $\boldsymbol{f_N}$ at one time step, the simulator can usually exploit coherence in L at the next time step. Although the coefficients given by $\mathbf{A}$ and $\mathbf{b}$ vary from one time step to the next, the index set L denoting a basic solution usually remains valid for some large number of time steps. Whenever L is known, $\boldsymbol{f_N}$ is easily found by computing $\mathbf{B}^{-1}\mathbf{d}$. As a result, $\boldsymbol{f_N}$ can be computed by solving only a linear equation for most time steps.

Using L from the previous time step, we may occasionally produce a singular $\mathbf{B}$ for the current time step or a non-positive vector $\mathbf{B}^{-1}\mathbf{d}$, indicating that L is invalid for the current time step. This usually happens when conditions change abruptly due to an impact. It also happens whenever an a_{Ni} becomes positive, indicating separation at a contact point. Since this may indicate a discontinuity in acceleration[30], the ordinary differential equation integrator is restarted at this point. Whenever L becomes invalid, a solution for $\boldsymbol{f_N}$ must be computed from scratch. The extra cost of forming $\mathbf{B}$ and $\mathbf{d}$ and computing $\mathbf{B}^{-1}\mathbf{d}$ even when L is invalid is still much less than the cost of always computing $\boldsymbol{f_N}$ from scratch. Gilmore and Cipra[20] discuss methods for predicting the change in L from one time step to the next.

The second implementation issue has to do with the formulation of $\boldsymbol{f_N}$ in terms of a QP. In analysis, it is well known that a necessary, but not sufficient condition for a function $y(x)$ to achieve a minimum at x_0 is that the gradient of y vanish at x_0. This is known as a *first order* optimality condition. For a QP, analogous first order conditions called the *Karush-Kuhn Tucker* (KKT) conditions for optimality exist. In the case of a convex QP, the KKT conditions are both necessary and sufficient;

the global minimum is achieved if and only if the KKT conditions hold[40]. For a PSD symmetric matrix $\mathbf{A}$, as is the case for frictionless configurations, the QP

$$\underset{\boldsymbol{f_N}}{\text{minimize}} \ \tfrac{1}{2}\boldsymbol{f_N}^T\mathbf{A}\boldsymbol{f_N} - \boldsymbol{f_N}^T\mathbf{b} \quad \text{subject to} \quad \boldsymbol{f_N} \geq \mathbf{0} \tag{8–11}$$

has KKT conditions

$$\mathbf{A}\boldsymbol{f_N} - \mathbf{b} = \boldsymbol{\lambda}, \quad \boldsymbol{f_N}^T\boldsymbol{\lambda} \geq 0 \qquad \text{and} \qquad \boldsymbol{\lambda} \geq 0. \tag{8–12}$$

(The elements of the n-vector $\boldsymbol{\lambda}$ are referred to as the *Lagrange multipliers* of the minimization problem.) A solution of Equation (8–11) is therefore a solution of Equation (8–10). The vector $\boldsymbol{f_N}$ may be computed by solving either Equation (8–10) or Equation (8–11). The form of the two different QP formulations will become more important when we deal with static friction in Chapter 14. In actual practice, Equation (8–10) is computationally preferable in dealing with frictionless systems. In dealing with the less-than-perfect world of numerical solution and optimization, including the explicit constraint $\mathbf{A}\boldsymbol{f_N} - \mathbf{b} \geq \mathbf{0}$ in the problem formulation increases the likelihood of finding an $\boldsymbol{f_N}$ that satisfies the most important constraint of non-penetration.

Chapter 9

The Coulomb Friction Model

At this point, we will extend our treatment of contact between bodies to the case when frictional forces act at contact points. We will call the force that arises between two bodies at a contact point a *contact force*. A contact force is the sum of a workless constraint force, and a friction force. The constraint force does no work on the system and acts normal to the contact surface; the constraint force is sometimes called the *normal force*. The friction force acts tangential to the contact surface, and may perform work by dissipating kinetic energy between contacting bodies. In this chapter, the model of friction used throughout the rest of this thesis is defined. Following this chapter, we will consider the theoretical and practical complexity of computing both normal and friction forces. We will say that the contact forces acting on a configuration with friction are valid if the friction forces satisfy the conditions defined in this section, and the constraint forces satisfy the conditions of the previous chapter.

9.1 Coordinate geometry at a contact point

At each contact point, let us define a local coordinate system as follows. Let $\hat{n}$, $\hat{t}_x$ and $\hat{t}_y$ denote mutually perpendicular unit vectors. The vector $\hat{n}$ is normal to the contact surface at the contact point while $\hat{t}_x$ and $\hat{t}_y$ span the plane tangent to the contact surface. As in Chapter 8, we adopt the convention that $\hat{n}$ points from B towards A; for example, $\hat{n}$ would be $\nabla G(p_b, \overline{p})/\|\nabla G(p_b, \overline{p})\|$ in Figure 6.4. We have already defined f_N and a_N, as the contact force and relative acceleration in the $\hat{n}$ direction. (In situations where we are discussing only a single contact point, we will drop subscripts and write f_N and a_N in place of $f_{N\,i}$ and $a_{N\,i}$.) Similarly, we decompose the friction force along the $\hat{t}_x$ and $\hat{t}_y$ axes into magnitudes f_x and f_y. Following the same convention as for normal forces, the friction force acting on A is $f_x\hat{t}_x + f_y\hat{t}_y$, with an equal and opposite friction force acting on B.

Since friction acts to oppose any slipping motion, let us consider the slip velocity tangent to the contact surface. In Equation (6–32) we derived s, the slip velocity between A and B at the contact point. Let s be projected onto the tangent plane and then decomposed along the $\hat{t}_x$ and $\hat{t}_y$ axes into magnitudes v_x and v_y:

$$v_x = \hat{t}_x \cdot s \qquad \text{and} \qquad v_y = \hat{t}_y \cdot s. \tag{9–1}$$

Thus, the tangential slip velocity (henceforth called the tangential velocity) is

$$v_x\hat{t}_x + v_y\hat{t}_y.$$

When the slip velocity s is zero, let the tangential slip acceleration be described by magnitudes a_x and a_y where

$$a_x = \hat{t}_x \cdot \frac{\mathrm{d}}{\mathrm{dt}}s \qquad \text{and} \qquad a_y = \hat{t}_y \cdot \frac{\mathrm{d}}{\mathrm{dt}}s. \tag{9–2}$$

If the tangential velocity is initially zero, then as long as the tangential acceleration

$a_x\hat{t}_x + a_y\hat{t}_y$ remains zero (and contact is not broken), the tangential velocity also remains zero.

The Coulomb model of friction is an accepted empirical relationship between the normal force magnitude f_N and the friction force magnitude $f_x^2 + f_y^2$. At all times, the condition

$$f_x^2 + f_y^2 \leq (\mu f_N)^2 \tag{9–3}$$

holds, where μ is a coefficient of friction that depends on material properties, and may be different at each contact point. If the tangential velocity is non-zero, we say that the friction force is *dynamic*; otherwise, the friction force is called *static*.

Typically, the coefficient μ of static friction (μ_{static}) is larger than the coefficient μ of dynamic friction ($\mu_{dynamic}$). Aside from this, μ is roughly independent of the tangential velocity. Since we are computing contact forces for a specific instant of time t_0, we know which coefficient of friction to use, and we will not bother to make a distinction between μ_{static} and $\mu_{dynamic}$. (We also will not bother to index μ over different contact points.)

9.2 Dynamic friction conditions

When dynamic friction arises, the friction force magnitude achieves its upper bound of μf_N. Also, the friction force acts directly opposite the slip velocity. Knowing that $f_x^2 + f_y^2 = (\mu f_N)^2$ and that the friction force is opposite the slip,

$$f_x\hat{t}_x + f_y\hat{t}_y = -\mu f_N \frac{v_x\hat{t}_x + v_y\hat{t}_y}{\sqrt{v_x^2 + v_y^2}} \tag{9–4}$$

or

$$f_x = -\mu f_N \frac{v_x}{\sqrt{v_x^2 + v_y^2}} \qquad \text{and} \qquad f_y = -\mu f_N \frac{v_y}{\sqrt{v_x^2 + v_y^2}}. \tag{9–5}$$

Thus, given μ and the tangential velocity, the dynamic friction force can be written

solely in terms of f_N. Since

$$\begin{aligned} f_N\hat{n} + f_x\hat{t}_x + f_y\hat{t}_y &= f_N\hat{n} + \frac{-\mu f_N v_x\hat{t}_x - \mu f_N v_y\hat{t}_y}{\sqrt{v_x^2 + v_y^2}} \\ &= f_N(\hat{n} + \frac{-\mu v_x\hat{t}_x - \mu v_y\hat{t}_y}{\sqrt{v_x^2 + v_y^2}}), \end{aligned} \tag{9–6}$$

the direction of the net contact force is predetermined given the tangential velocity and μ.

9.3 Static friction conditions

The conditions between f_x, f_y and f_N are more complex when the tangential velocity is zero, resulting in static friction. There are two possibilities, according to whether or not the tangential acceleration is zero. First, f_x and f_y may satisfy

$$a_x = a_y = 0 \qquad \text{and} \qquad f_x^2 + f_y^2 \leq (\mu f_N)^2, \tag{9–7}$$

indicating that the friction remains static. Second, f_x and f_y may satisfy

$$f_x^2 + f_y^2 = (\mu f_N)^2 \qquad \text{and} \qquad (f_x\hat{t}_x + f_y\hat{t}_y) \cdot (a_x\hat{t}_x + a_y\hat{t}_y) \leq 0, \tag{9–8}$$

indicating that slipping is occurring, and that the static friction force has become dynamic. In this case, the friction force must oppose the initial tangential acceleration and have magnitude μf_N. The friction force direction does not have to be directly opposite the tangential acceleration direction.

9.4 Impulsive frictional forces

When two bodies collide at a contact point with friction, we can consider the collision as taking place over some small but non-zero time interval. We then consider the

limiting value of the product of the force acting between the bodies and the time interval, as we let the time interval tend to zero. This limiting value is the impulsive force of the collision. During the time interval of the collision, the normal and friction forces must satisfy the conditions of the previous two sections. It is possible that during this time interval, the direction of the relative tangential velocity between the bodies may change. For planar collisions, this happens if the relative tangential velocity passes through zero and reverses itself. Wang and Mason[34,54] describe the computation of the impulsive forces for all possible planar collisions involving one contact point. For three-dimensional collisions, the velocity direction can vary in the tangent plane of the collision, and the computational problem becomes much more difficult.

For simulation purposes, the simulator described in this thesis makes the simplification that the direction of the relative tangential velocity does not change during a collision. For planar collisions, this amounts to ignoring the possibility of reversed sliding, as considered by Wang and Mason. For three-dimensional collisions, this is simply an out-and-out approximation that is reasonable for some situations, and probably unreasonable for others. This model is no more than an *ad hoc* approximation that allows the simulator to produce a (hopefully) reasonable looking collision response behavior. However, the model, though incorrect physically, is computationally attractive because it produces precisely the same sort of constraint equations as are encountered for static friction.

The idea is the following. Given a collision at a contact point, let f_N denote the normal impulse magnitude, f_x and f_y the tangential frictional impulse, and v_N and v_N^+ the relative approach speeds in the the normal direction prior to the collision, and after the collision, respectively. Similarly, let v_x^+ and v_y^+ denote the tangential velocity components after the collision. The constraints on the normal quantities are $f_N > 0$, and $v_N^+ = -\epsilon v_N$ where ϵ is the coefficient of restitution of

the collision. Note that this kinematic condition, which is Newton's hypothesis, is known to violate energy conservation under some circumstances if both ϵ and μ are non-zero[3]. The standard condition $f_x^2 + f_y^2 \leq (\mu f_N)^2$ is imposed at all times on the frictional impulse.

If the bodies are sliding after application of the impulse, then the frictional impulse must achieve its upper bound, and must oppose the sliding velocity. This is written as

$$f_x^2 + f_y^2 = (\mu f_N)^2 \qquad \text{and} \qquad (f_x \hat{t}_x + f_y \hat{t}_y) \cdot (v_x^+ \hat{t}_x + v_y^+ \hat{t}_y) \leq 0. \tag{9–9}$$

Otherwise, the bodies are not sliding, and the frictional impulse need only satisfy its upper bound condition. This yields

$$v_x^+ = v_y^+ = 0 \qquad \text{and} \qquad f_x^2 + f_y^2 \leq (\mu f_N)^2. \tag{9–10}$$

Comparing these last two equations to Equations (9–8) and (9–7), it is clear that both sets of equations can be solved by the same techniques. Accordingly, we will be able to use the solution methods described in Chapter 14 to compute frictional impulses.

If we are dealing with a number of contact points simultaneously, we will need to add the following kinematic constraint. If at some colliding contact $v_N^+ > -\epsilon v_N$, then the normal impulse at that contact must be zero. In other words, if bodies bounce away at a contact point with a speed in excess of that predicted by the coefficient of restitution, the normal impulse at that point must be zero[13]. Satisfying these conditions is still exactly analogous to satisfying the static friction conditions for a collection of non-colliding contact points, so again, we can compute frictional impulses using the static friction methods described in Chapter 14.