MULTIBODY DYNAMICS 2003
Jorge A.C. Ambrósio (Ed.)
IDMEC/IST, Lisbon, Portugal, July 1–4, 2003

AN ALGORITHM FOR COMPLIANT CONTACT BETWEEN COMPLEXLY SHAPED SURFACES IN MULTIBODY DYNAMICS

Gerhard Hippmann
Institute of Aeroelasticity
DLR – German Aerospace Center, Oberpfaffenhofen
Münchnerstr. 20, D-82234 Wessling, Germany
e–mail: pcm@hippmann.org, web page: http://www.pcm.hippmann.org

Keywords: Multibody Dynamics, Contact Mechanics, Computer Graphics, Collision Detection, Computer Aided Engineering

Abstract. *A new contact algorithm designed for multibody dynamics is presented. It is based on representation of the body surfaces by polygon meshes and contact force determination by the elastic foundation model. Areal discretisations of the contact patches are constructed using methods closely related to computer graphics, e.g. collision detection based on bounding volume hierarchies and generation of subdivision surfaces by means of boundary representation data structures. Two examples prove the robustness of the method for complexly shaped bodies causing multiple and multiply bordered contact patches and conforming contacts.*

1 INTRODUCTION

Representation of contact mechanics adequate for multibody dynamics is still a big challenge. It is eminently difficult to find methods and algorithms which can model the highly complex phenomenon of contacting bodies realistically and efficiently enough for MBS simulation.

A widespread approach is to reduce the contact problem to a geometrically determined contact point and normal direction [1, 2]. As a rule these methods can be divided into two groups [3]: On the one hand rigid bodies are postulated resulting in unilateral constraints and hypotheses of impact. On the other hand specific force elements representing unilateral spring–damper force laws are used.

Both types have been implemented several times and have produced satisfactory results for many applications. However assuming contact points has at least two substantial weak points:

First it is in general not sufficient to examine a contact problem in a single point of the contact area which can result in essential errors of the approximation. Secondly the contact point determination is costly or even impossible for complexly shaped body surfaces with concave sections leading to multiple or conforming contact.

The Finite Element Analysis is doubtless the most powerful numerical method for solving contact problems. Eberhard [4] deals with FEA contact analysis within MBS simulation in detail. This approach appears to be suitable for particularly high requirements of accuracy. But its typical computational effort for treating three–dimensional, dynamic contact problems exceeds the range of multibody dynamics by several orders of magnitude.

This paper presents a contact algorithm which belongs to another class, in which the contact is examined using an areal discretisation of the contact patch. It can therefore be regarded as a compromise between the simple contact point approach and the costly FEA and has been designed with the intention to avoid the drawbacks of these methods.

In contrast to other algorithms of this class [5, 6, 7, 8], the Polygonal Contact Model (PCM) can treat polygonal body surfaces of arbitrary complexity and properly implements the elastic foundation model.

2 FUNDAMENTALS

PCM has two essential characteristics: The surfaces of the bodies are described in polygonal representation and the contact force determination is based on the elastic foundation model. This chapter briefly introduces these fundamental terms.

2.1 Polygonal surfaces

In Virtual Reality (VR) and Computer Aided Design (CAD) many methods established for representing freeform surfaces, e.g. Bezier patches and Non Uniform Rational B–Splines (NURBS). A comparatively simple alternative is to approximate them by polygon meshes. Polygonal surfaces consist of vertices and faces. A vertex is defined by its position coordinates in the Euclidean space. A face is defined by pointers to the vertices of a polygon. The totality of its faces represents the topology of a polygonal surface.

Polygon meshes are used as internal representation of surfaces in computer graphics. Therefore most programs dealing with freeform surfaces are equipped with appropriate export filters. Besides, there are plenty of publications and software tools dealing with polygonal surfaces. And many Internet web pages provide free or commercial libraries of 3D–meshes of various areas.

In practice the quality of polygonal surfaces differs considerably. PCM makes quite high demands on meshes representing body surfaces: They must not contain duplicate vertices or cracks and their polygons have to be oriented consistently. Moreover their resolution essentially influences the discretisation of the contact patch (see Section 3.2). In the present implementation

PCM requires polygonal surface files in the simple Wavefront format [9]. Quadrangles and pentagons are converted into two respectively three triangles when loading the file.

2.2 Elastic foundation model

The physical phenomena determining contact dynamics are exceedingly complex. Therefore contact models have to abstract reality quite extensively to be well suited to typical MBS simulations concerning efficiency. In an extensive literature research the elastic foundation model came off to promise a good balance of numerical expense and physical quality [10].

Just like classical multibody dynamics the elastic foundation model postulates rigid bodies, but they are assumed to be covered by thin elastic layers. Neglecting tangential share stress in a layer of thickness b results in a direct relation of normal displacement u_n and pressure p_n [11]:

$$p_n = \frac{K}{b} \cdot u_n \tag{1}$$

For thin, linear–elastic layers characterised by their Young's modulus E and their Poisson's ratio $\nu < 0.45$ the elastic modulus K is given by [12]:

$$K = \frac{1-\nu}{(1+\nu)(1-2\nu)} \cdot E \tag{2}$$

Most applications do not meet the assumption of thin layers. However the elastic foundation model can also be applied successfully to homogeneous, elastic bodies if the layer stiffness

$$c_l = \frac{K}{b} \tag{3}$$

is determined by experiments or qualified estimating – similar to the coefficient of restitution of rigid body contact.

PCM implements the elastic foundation model based on a discretisation of the contact patch. In every triangular contact element normal displacement and pressure are assumed to be constant. So according to (1) the normal force of a contact element k of area A_k results to:

$$F_{nk} = c_l \cdot A_k \cdot u_{nk} \tag{4}$$

Note that the linear–elastic relation (1) can easily be modified to model other rheological properties, e.g. visco–elastic material. Besides, the discretisation of PCM might also be used to implement the powerful half–space approximation [13].

3 POLYGONAL CONTACT MODEL

For contact analysis PCM provides three tasks with corresponding software interfaces. Before simulation Task 1 defines a contact pairing and executes corresponding preprocessing steps

including loading of the surface files, calculation of dependent parameters and generation of geometrical data structures. Task 2 performs the contact analysis of a contact pairing for a given system state. After simulation Task 3 disposes PCM's memory.

From the MBS point of view PCM behaves like an ordinary force element. On each body of a contact pairing a marker frame M with constant position $\mathbf{r}_{RM}$ and orientation A_{RM} towards its reference frame R is to be defined. PCM utilises these markers M_e, M_f as reference frames of the polygon meshes representing the bodies' surfaces E, F (Fig. 1). For the contact analysis task PCM requires the current relative kinematics (position vector $\mathbf{r}_{M_eM_f}$, orientation matrix $A_{M_eM_f}$, velocity vectors $\mathbf{v}_{M_eM_f}$ and $\omega_{M_eM_f}$) of the markers and returns contact force F_{M_e} and torque T_{M_e} to be applied on body i at the location of marker M_f.

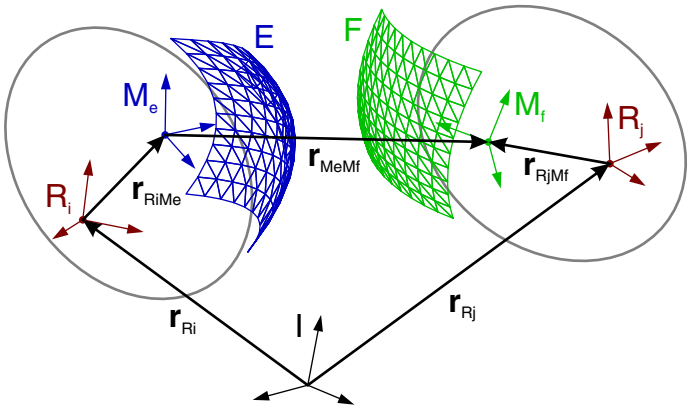

Figure 1: Kinematic terms of a contact pairing

The PCM contact analysis task consists of three steps. First a collision detection algorithm determines if the contact pairing is in touch. If no collision is detected, the program returns zero force and torque and the analysis is finished. Otherwise PCM constructs in the second step the intersecting areas of the surfaces and discretises the corresponding contact patches. Finally the contact force of each contact element is determined and the resulting contact force and torque of all contact elements is calculated.

3.1 Collision detection

Collision detection is a sophisticated discipline of VR [14]. The basic problem is to find out if two surfaces intersect for a given relative position and orientation. PCM implements an exact and highly efficient algorithm based on Bounding Volume (BV) hierarchies [15].

Two polygonal surfaces collide if at least one pairing of intersecting polygons exists. Therefore the basic operation of all exact collision detection algorithms is an intersection test for two polygons in 3D–space. PCM's collision detection includes an efficient algorithm specialised in

triangles [16] which requires between 76 and 181 floating point operations (FLOPs) per evaluation (including vertex transformations).

The simplest conceivable method is to test each triangle of surface E against each triangle of surface F for intersection. But this brute force approach results in unacceptable calculation effort since its complexity grows quadratically with the number of triangles of the surfaces.

The strategy of BV–based collision detection is to avoid as many polygon intersection tests as possible. This is achieved by accessing a BV hierarchy calculated once for every surface during preprocessing.

The BVs utilised by PCM are cuboids fixed and axis–aligned to the surface's reference frame. Each BV represents a convex hull of contiguous parts of the surface. The totality of BVs forms a binary tree hierarchy in which the BV–size decreases with increasing tree level.

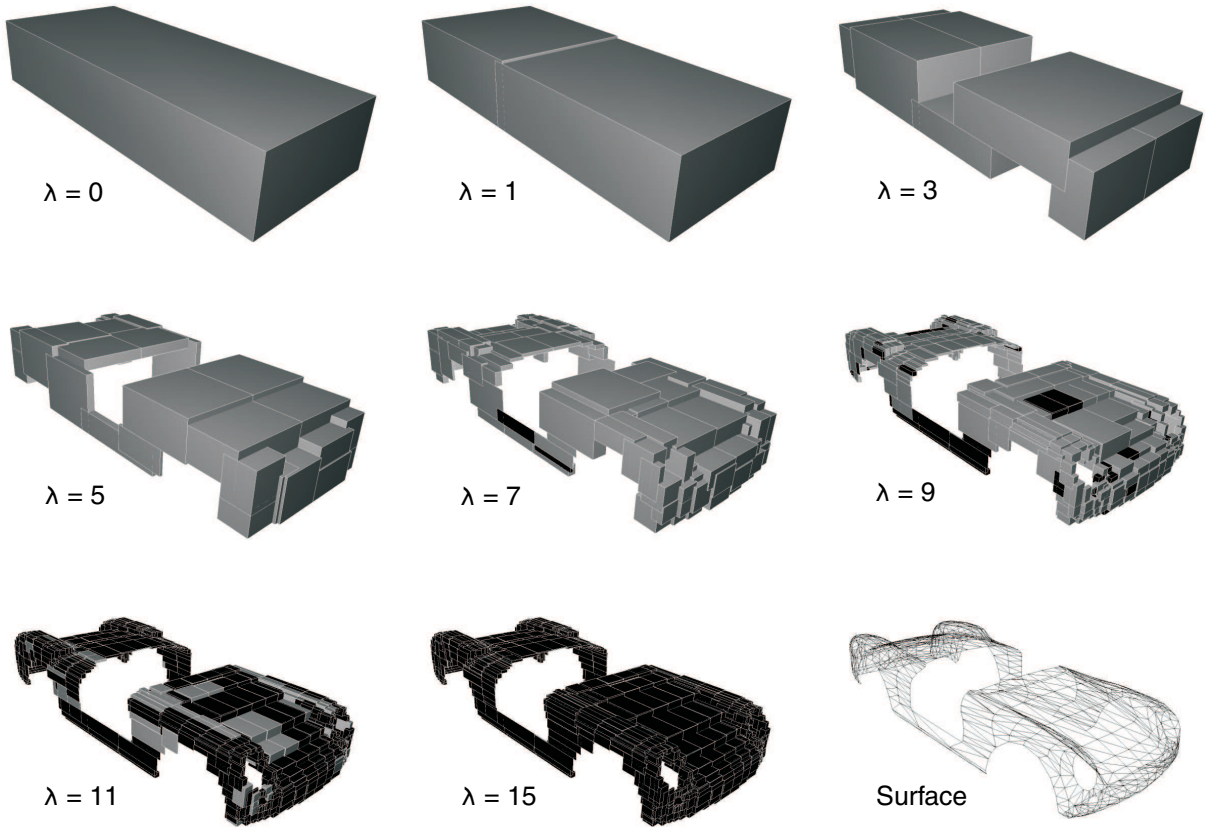

Figure 2: Several levels λ of a Bounding Volume Hierarchy of a car body surface

Fig. 2 illustrates several levels of a BV hierarchy. The root element includes the whole surface. From one level to the next all BVs are gradually divided into two child BVs, each including half of their triangles. Thereby the greatest spatial expanse determines the axis direction of the

subdivision. Finally, the leaf elements of the tree (marked black in Fig. 2) include one solitary triangle. PCM implements a recursive algorithm to calculate BV–trees.

During simulation the collision detection is performed at the beginning of every contact analysis for the current relative position $\mathbf{r}_{M_eM_f}$ and orientation $A_{M_eM_f}$. The marker frame M_e corresponding to surface E is used as the reference frame of all operations and vectors without loss of generality.

The evaluation starts with a collision test of the root level BVs of the two surfaces. If an intersection is detected the four possible collision tests between the child elements are gradually performed. Otherwise the current BV pairing and all its children are excluded from the further evaluation because their included triangles cannot intersect.

Usually only small areas of the surfaces are in contact. Therefore this recursive algorithm finds the intersecting leaf element BVs by a small number of simple collision tests between cuboids (12 to 36 FLOPs). Consequently the costly triangle intersection test is to be performed for the few triangle pairings with colliding BVs only (Fig. 3).

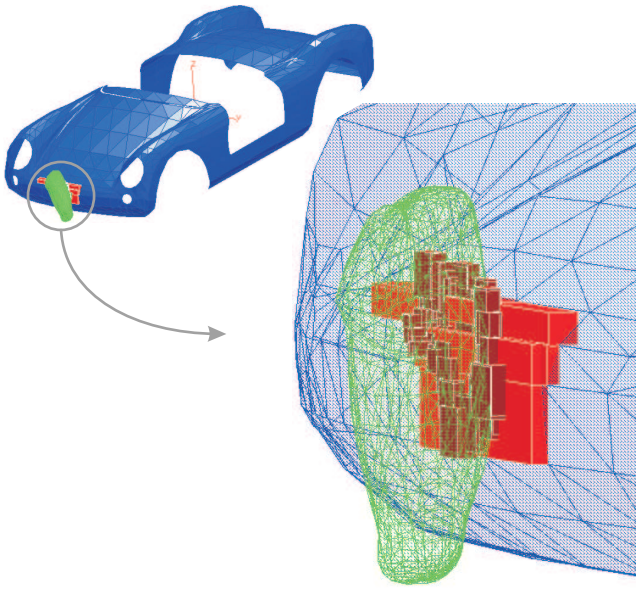

Figure 3: Collision detection between a car body and an upper leg surface: The BV–algorithm reduces the number of costly triangle intersection tests from 1890336 to 330 plus 1537 simple BV collision tests

In a comparison PCM's collision detection calculated about 10^4 times faster than the brute force method with identical results. As a consequence it normally adds negligible numerical effort to the contact analysis.

3.2 Contact element generation

The basic idea of PCM is to approximate the contact patch and subsequently divide it into contact elements which behave similarly to force elements of the contact point approach. The concept to determine this discretisation differs considerably from kindred contact algorithms since it is closely related to computer graphics.

The contact element generation consists of three steps. In Step 1 the intersection polygon is constructed, Step 2 stands for determination of the active areas of the body surfaces and in Step 3 the contact elements are derived from the active areas.

The intersection of two polygonal surfaces is a three–dimensional polygon (Fig. 4). Every intersection line belongs to one polygon of each of the both of the surfaces. The collision detection returns all intersection lines including the IDs of the intersected triangles and the position vectors of its end points in irregular order.

Figure 4: Intersection Polygon of a car body and a (halved) upper leg surface

The construction of the intersection polygon starts with the first line of the stack. Then the Euclidean distances between its end points and all end points of the remaining lines are calculated. If the minimal distance is less than a certain snap distance (well–tried value: $10^{-20}m$) the corresponding line is appended to the intersection polygon and the procedure is gradually rerun for its end points. Otherwise it is taken as the first line of another intersection polygon constructed in the same way. At the end of Step 1 PCM closes the intersection polygons if the distance of their end points is less than the snap distance.

Note that multiple or multiply bordered contact patches (Fig. 8) result in multiple intersection polygons. Besides open surfaces can lead to open intersection polygons if bordering edges appear in the contact area.

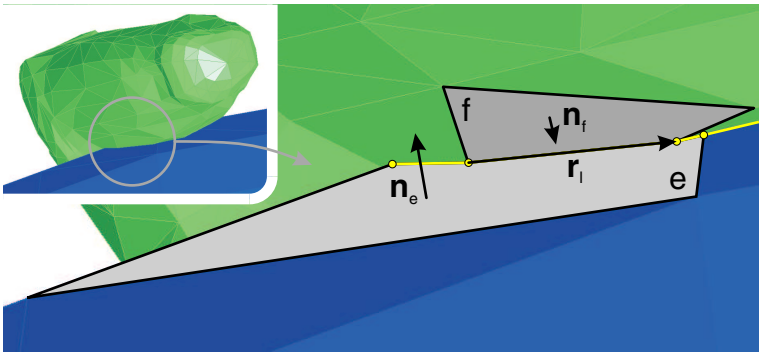

Figure 5: Determination of the circulation direction of an intersection polygon

In preparation for the following Step 2 a circulation direction is to be defined for each intersection polygon. Therefore an arbitrary intersection line and its triangles are considered (Fig. 5). The direction of the line vector $\mathbf{r}_l$ is chosen so that it forms a right hand system with the normal vectors $\mathbf{n}_e, \mathbf{n}_f$ of the intersected triangles of the surfaces E and F:

$$(\mathbf{n}_e \times \mathbf{n}_f) \cdot \mathbf{r}_l > 0 \tag{5}$$

In doing so the normal vectors are assumed to point to the outside of the bodies. Thus if you imagine to go along the intersection polygon on surface E the intersection volume always lies on your left hand side.

In Step 2 the active areas of the surfaces are determined. Every intersection consists of two active areas, one of surface E and the other one of surface F. An active area is a subdivision of a surface formed by all intersected and inner triangles of an intersection which form the boundary of the intersection volume (Fig. 6).

The determination of the active areas is based on boundary representation (B–Rep) data structures known as Doubly Connected Edge Lists (DCEL) [17] which are generated for every surface during preprocessing. A DCEL consists of data elements providing pointers to the adjacent faces, vertices and edges of each edge of a polygonal surface (Fig. 7). In PCM it is used to follow up the edges of the active areas to find their inner faces and vertices. Note that a workable DCEL can only be generated for surfaces without duplicate vertices or cracks.

The search for inner faces is performed for every edge intersected by an intersection polygon. At first the inner vertex is determined using a geometrical relation similar to (5). Then all its adjacent faces and vertices are searched DCEL–based. If a candidate vertex is not yet part of the active area, the adjacent faces and the vertex itself are added to it and the search is continued gradually for all adjacent edges. Otherwise the current branch of the recursive algorithm is terminated.

For multiply bordered contact patches (Fig. 8) the search procedure becomes more complicated. Before a new vertex can be added all remaining intersection polygons have to be checked

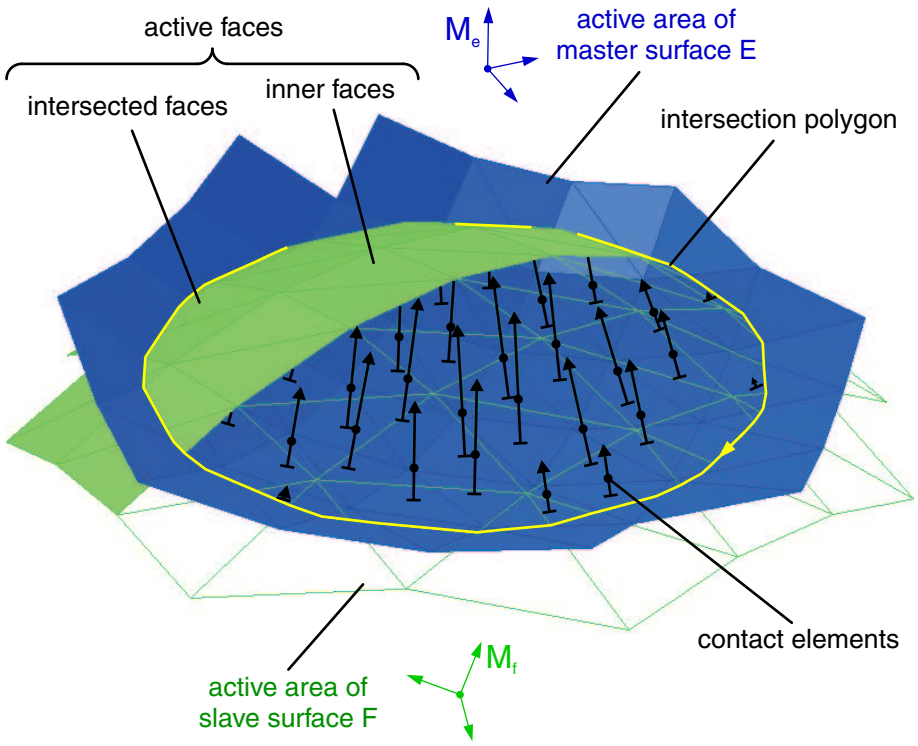

Figure 6: Components of an intersection. The contact elements are positioned perpendicular at the barycenters of the active faces of the master surface

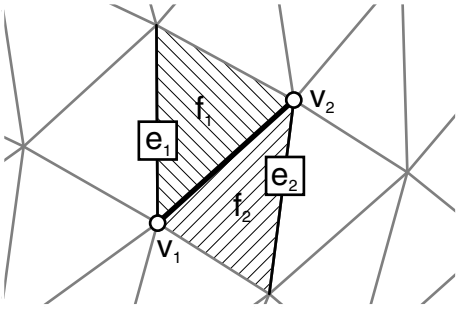

Figure 7: A Doubly Connected Edge consists of pointers to its adjacent faces, vertices and edges

for intersection with the current edge. If an intersection is detected the candidate vertex must not be added to the active area, the search branch has to be terminated and the other polygon with all its intersected triangles has to be added to the current intersection.

The last action of Step 2 is to remove duplicate faces and vertices of the active areas which may occur for multiple intersections (Fig. 8). After that the generation of the contact elements begins.

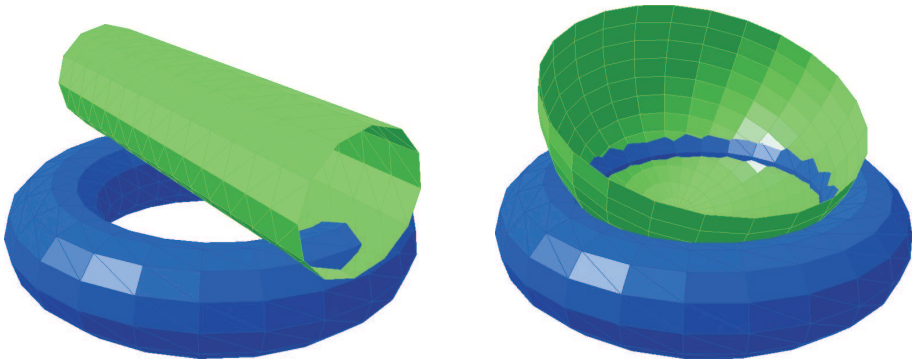

Figure 8: Contacts with multiple intersections (left) and a multiply bordered contact patch (right)

PCM distinguishes between three modes of contact element generation. In the first mode surface E is treated as master and F as slave, and in the second F is master and E slave. In the third one both variants are performed successively and the resulting contact force and torque vectors are calculated as average values. In the following only the first mode is considered.

The elastic foundation model is based on the penetration u_n of the undeformed surfaces of the contacting bodies. In the discretised version (4) of PCM u_{nk} is the penetration of a contact element which is defined by three additional attributes: Its area A_k, its unified normal vector $\mathbf{n}_k$ and its position C_k (Fig. 9).

PCM generates one contact element for each active triangle e of the master surface E. For this reason the contact patch discretisation results from the resolution of E so that the master/slave–mode of the contact element generation has to be chosen carefully.

The area A_k is set identical to the master triangle:

$$A_k := A_e \tag{6}$$

The normal vector $\mathbf{n}_k$ is also adopted of the master triangle, but its sign is adapted so that it points to the inside of body E (according to the resulting contact force):

$$\mathbf{n}_k := -\mathbf{n}_e \tag{7}$$

As penetration u_n the Euclidean distance from the master triangle's barycenter C_e to the intersection point I of its barycenter–normal

$$\mathbf{r} = \mathbf{r}_{M_e C_e} + d\,\mathbf{n}_e \;\; (d \in \mathbb{R}) \tag{8}$$

with the slave surface F is determined. For that purpose PCM searches for the intersected slave triangle f of the active area of F as follows.

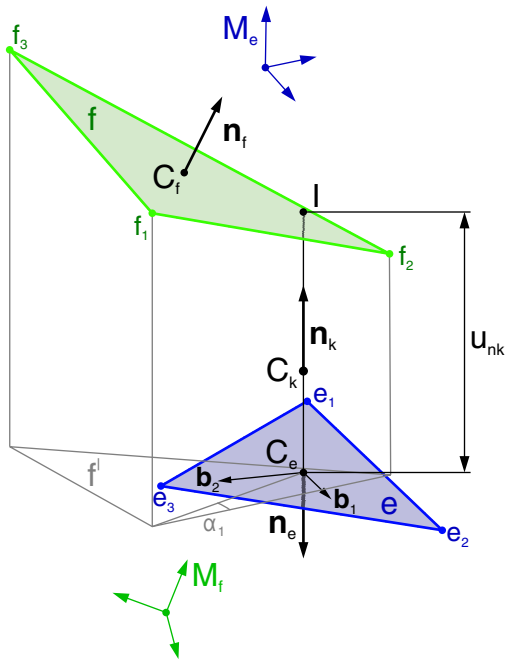

Figure 9: Geometric quantities of the contact element generation

First a local basis $(\mathbf{b}_1, \mathbf{b}_2)$ of e is calculated by

$$\begin{aligned} \mathbf{b}_1 &:= \frac{\mathbf{r}_{M_e e_2} - \mathbf{r}_{M_e e_1}}{|\mathbf{r}_{M_e e_2} - \mathbf{r}_{M_e e_1}|} \\ \mathbf{b}_2 &:= \mathbf{n}_e \times \mathbf{b}_1 \end{aligned}$$

where $\mathbf{r}_{M_e e_i}$ is the position vector of a vertex e_i $(i \in [1,2,3])$ of e. To decide if the straight line (8) intersects a slave triangle f the problem is considered two–dimensionally in the plane $(\mathbf{b}_1, \mathbf{b}_2)$ of e. Therefore the vertices f_i of each candidate f are projected by:

$$\mathbf{r}'_{C_e f_i} := [\mathbf{b}_1\ \mathbf{b}_2]^T \left(\mathbf{r}_{M_e f_i} - \mathbf{r}_{M_e C_e}\right) \tag{9}$$

Consequently the normal (8) is known to intersect f if the barycenter C_e lies on the inside of f'. This is fulfilled if all angles α_i between the edge vectors $\mathbf{r}'_{f_i f_{i+1}} = \mathbf{r}'_{C_e f_{i+1}} - \mathbf{r}'_{C_e f_i}$ and the reversed vertex position vectors $\mathbf{r}'_{f_i C_e} = -\mathbf{r}'_{C_e f_i}$ are positive. Hence for the intersected slave triangle f holds

$$\det\left[\mathbf{r}'_{f_i f_{i+1}}\ \mathbf{r}'_{f_i C_e}\right] > 0 \quad \forall i \in [1,2,3]\,. \tag{10}$$

After f is found the penetration u_{nk} of the contact element is determined using (8) and the normal form of the plane of f:

$$u_{nk} := \frac{\left(\mathbf{r}_{M_e C_f} - \mathbf{r}_{M_e C_e}\right) \cdot \mathbf{n}_f}{\mathbf{n}_e \cdot \mathbf{n}_f} \tag{11}$$

At last the position C_k of the contact element is to be determined. This point approximates the location of the contact patch and represents the reference position of the relative velocity vector $\mathbf{v}_k$ of the surfaces and of the resulting contact force vector $\mathbf{F}_k$.

In general it is quite costly to find the actual shape of the contact patch of elastic bodies in contact [12]. PCM uses a slightly extended version of the elastic foundation model that allows a straightforward approximation: Both contacting bodies E and F are assumed to be covered by elastic layers of moduli K_E, K_F and thicknesses b_E, b_F. Consequently a serial connection of (1) results in the combined layer stiffness

$$c_l := \frac{c_E \cdot c_F}{c_E + c_F} = \frac{1}{b_E/K_E + b_F/K_F}. \tag{12}$$

Accordingly the ratio of penetration of the layers leads to the contact element position

$$\mathbf{r}_{M_eC_k} := \mathbf{r}_{M_eC_e} + \frac{c_F}{c_E + c_F} \cdot u_{nk} \cdot \mathbf{n}_k. \tag{13}$$

3.3 Contact force determination

The determination of the contact force vector $\mathbf{F}_k$ of each contact element is similar to force elements designed for the single contact point approach. The normal force is composed of the elastic share

$$F_{ck} := c_l \cdot A_k \cdot u_{nk} \tag{14}$$

according to the elastic foundation model (4) and an analogously defined viscous damping share

$$F_{dk} := \begin{cases} d_l \cdot A_k \cdot v_{nk} & : \quad u_{nk} \geq u_d \\ d_l \cdot A_k \cdot v_{nk} \cdot \frac{u_{nk}}{u_d} & : \quad u_{nk} < u_d \end{cases} \tag{15}$$

determined by the areal layer damping factor d_l and the normal share

$$v_{nk} = \mathbf{n}_k \cdot \mathbf{v}_k \tag{16}$$

of the rigid bodies' relative velocity vector

$$\mathbf{v}_k = \mathbf{v}_{M_eM_f} + \omega_{M_eM_f} \times \left(\mathbf{r}_{M_eC_k} - \mathbf{r}_{M_eM_f}\right) \tag{17}$$

at the contact element position C_k. In (15) a linear transition fades out the damping force if the penetration is less than a given transition depth u_d. This empirical approach can be used to avoid discontinuous contact forces at the beginning of collisions.

When calculating the total normal force F_{nk} another empirical strategy prevents unrealistic tension forces:

$$F_{nk} := \begin{cases} F_{ck} + F_{dk} & : \quad F_{ck} + F_{dk} > 0 \\ 0 & : \quad F_{ck} + F_{dk} \leq 0 \end{cases} \tag{18}$$

The tangential force of the contact element is determined dependent on the tangential relative velocity

$$v_{tk} = |\mathbf{v}_k - v_{nk} \cdot \mathbf{n}_k| \tag{19}$$

and its normal force F_{nk}:

$$F_{tk} := \begin{cases} \mu \cdot F_{nk} & : \quad v_{tk} \geq v_\varepsilon \\ \mu \cdot F_{nk} \cdot \frac{v_{tk}}{v_\varepsilon} \left(2 - \frac{v_{tk}}{v_\varepsilon}\right) & : \quad v_{tk} < v_\varepsilon \end{cases} \tag{20}$$

This is a regularised version of Coulomb's law of dry friction (Fig. 10): If the slip velocity falls below a small limit v_ε the friction force is faded out quadratically to avoid the set–valued state of static friction.

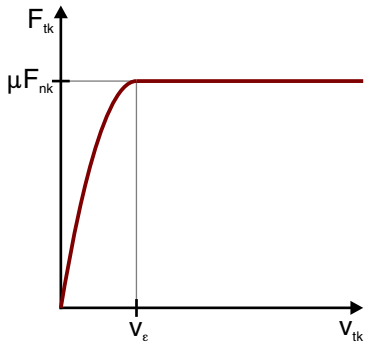

Figure 10: Regularised version of Coulomb's friction law

The total force vector $\mathbf{F}_k$ of the contact element is given by a projection of the normal component F_{nk} along its normal vector $\mathbf{n}_k$ and of the tangential component F_{tk} counteracting the direction of the tangential velocity vector $\mathbf{v}_{tk}$:

$$\mathbf{F}_k = F_{nk} \cdot \mathbf{n}_k + F_{tk} \cdot \frac{\mathbf{v}_{tk}}{v_{tk}} \tag{21}$$

According to the conventions of MBS software interfaces PCM determines the contact force acting on body E at the position of marker M_f. Therefore the resulting torque vector $\mathbf{M}_k$ of a contact element is given by:

$$\mathbf{M}_k = \left(\mathbf{r}_{M_e C_k} - \mathbf{r}_{M_e M_f}\right) \times \mathbf{F}_k \tag{22}$$

Finally the forces and torques of all contact elements are summed up resulting in the total force and torque of the contact pairing:

$$\mathbf{F}_E^{M_f} = \sum_k \mathbf{F}_k \tag{23}$$

$$\mathbf{M}_E = \sum_k \mathbf{M}_k \tag{24}$$

Obviously PCM represents a classical force element which does not introduce state variables, algebraic constraints or root functions in the multibody system, but returns applied force and torque vectors for a given set of relative kinematics on position and velocity level. This simplicity appeared to be a valuable advantage in practical use.

4 APPLICATIONS

So far PCM has been applied extensively to a couple of test models. The two following examples were designed to prove the correct implementation of PCM. All input data traces back to reasonable suppositions and estimations. Nevertheless a quantitative verification of PCM has yet to come.

4.1 Bouncing bubbles

Bouncing bubbles is a synthesised model consisting of three identical bubbles bouncing in a pyramidal container due to their weights. The rotationally symmetric geometry of the bubbles is designed to bring about multiple and conforming contacts which cannot be handled reliably by contact models based on single point contact. Every bubble is $40\ cm$ long, weighs $2\ kg$ and is represented by a polygonal surface of 950 triangles. The inertially fixed container is represented by four triangles.

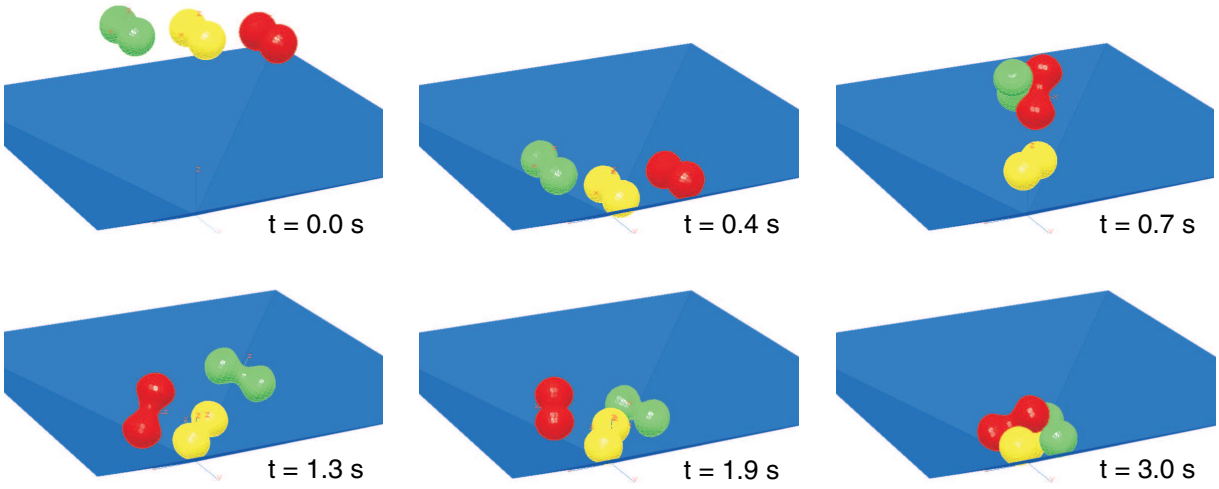

Figure 11: Snapshots of the bouncing bubbles simulation

The MBS has 18 degrees of freedom resulting in an ODE with 36 first order states. There are three PCM contact pairings between the bubbles ($c_l = 5.9 \cdot 10^5\ N/m^3, d_l = 1000\ Ns/m^3, \mu = 0.25$) and another three between the bubbles and the container ($c_l = 1.2 \cdot 10^6\ N/m^3, d_l = 500\ Ns/m^3, \mu = 0.25$). For an initial height of fall of $1\ m$ the bubbles bounce for about three

seconds (Fig. 11) whereby the solution is quite sensitive to physical and numerical parameters. Table 1 summarises the statistics of the simulation for the ODE integrator LSODE [19] and the DAE integrator DASSL [18] for absolute and relative tolerances $atol = rtol = 2.0 \cdot 10^{-4}$.

simulation–time 3.0 s	LSODE	DASSL
mean step–size	0.8 ms	1.2 ms
function evaluations	6046	7100
Jacobian evaluations	—	88
CPU–time[1]	276 s	286 s
real–time factor[1]	92.1	95.3

Table 1: Integration statistics of the bouncing bubbles simulation

4.2 Pedestrian crash

An essential motivation to the development of PCM were the grave problems in simulating a crash of a car and a pedestrian with the single contact point approach [20]: A detailed representation of the car body could not be modeled since the surface approximation has to be continuously differentiable up to the second order and in most simulations the contact point determination failed due to concave sections of the surfaces involving multiple contact points, contact point jumps and conforming contacts. For this reason a pedestrian crash was the first large test model for PCM (Fig. 12).

The man model consists of 15 segments linked by spherical, cardan and rotational joints resulting in an MBS with 29 degrees of freedom (58 first order states). The inertial properties of the segments are based on the parameterisation formulas by Dempster [21] for a total mass of $75\ kg$. All 15 joints of the man are constrained in their angular ranges by force elements representing linear–elastic rotational bumpers. The car is moved along the road with constant velocity of $30\ km/h$.

The polygonal surfaces of the man segments and the car body have been taken from free Internet libraries [22, 23] and were worked up with the shareware tool AC3D [24]. There are defined 15 contact pairings between the car and the man ($c_l = 3.2 \cdot 10^7\ N/m^3, d_l = 1.0 \cdot 10^5\ Ns/m^3, \mu = 0.4$) and 12 additional pairings between segments of the man ($c_l = 2.1 \cdot 10^7\ N/m^3, d_l = 2.0 \cdot 10^5\ Ns/m^3, \mu = 0.4$). The contact surfaces of all segments consist of 20452 triangles and the car body of 1240 triangles.

Of course the linear–elastic foundation model is a quite bad approximation for the complex elastic properties of the parts of a human body and the inhomogeneous stiffness distribution of a car body. A more realistic model could be obtained by replacing the spring–damper based force law of the contact elements by a more suitable rheological configuration including nonlinear, visco–elastic and plastic characteristics.

[1] Mobile PIII 1133 MHz, SIMPACK 8.6, Win 2k

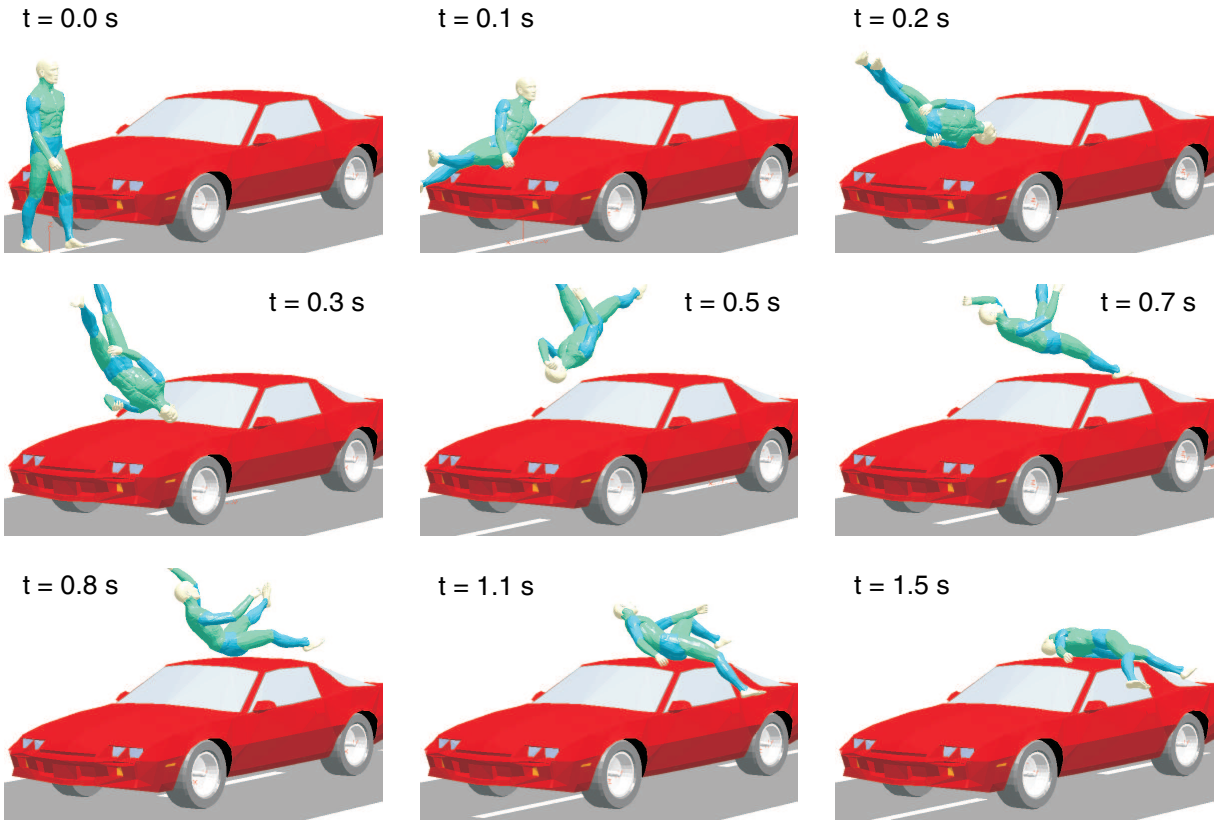

Figure 12: Snapshots of the pedestrian crash simulation

simulation–time 1.5 s	DOPRI5	DASSL
mean step–size	2.0 ms	0.8 ms
function evaluations	4501	5020
Jacobian evaluations	—	30
CPU–time[1]	288 s	336 s
real–time factor[1]	192	224

Table 2: Integration statistics of the pedestrian crash simulation

Table 2 shows the time integration statistics for DOPRI5 [25] and DASSL ($atol = rtol = 1.0 \cdot 10^{-3}$). In the dynamical phase from 0.0 to $1.5\ s$ of the simulation PCM performs efficiently. Afterwards the man remains lying on the car roof nearly motionless, but surprisingly the integration slows down notably. Supposably this effect is caused by the regularised range of the friction force law (20). This problem is to be analysed in the future.

After several enhancements the algorithm turned out to be absolutely robust even though it

includes some geometric operations which could fail for singular cases (e.g. (11)). In hundreds of test simulations no error occurred. Multiple and multiply bordered contact patches as well as conforming contacts were processed without any problems. Altogether PCM's numerical efficiency appears to be quite acceptable in consideration of its advantages in practical use.

A significant drawback of PCM is that it requires several parameters which are not available in real life, namely the layer stiffness c_l and damping factor d_l and the regularisation velocity v_ε. Hence parameter identification based on experiments or a lot of operational experience is needed for the use in practical engineering. But note that this is also valid for force laws and hypotheses of impact used in the single point contact methods.

The easy availability of polygonal contact surfaces is of great advantage. But if they have bugs like cracks, duplicate vertices or inconsistently oriented faces it can be laborious to work them up for PCM. Furthermore numerical accuracy and efficiency of PCM depend strongly on the resolution of the surfaces because the discretisation of the contact patch results from the triangulation of the master surface. For this reason the resolution and the master/slave–mode of the contact element generation have to be chosen carefully.

Since PCM extensively utilises dynamically allocated data structures it was implemented in C. The current state of the source code is available from the PCM website [26], together with interface routines and test models for SIMPACK. However it should be quite easy to link the code to other MBS–programs.

5 CONCLUSIONS

A new method for analysing contacts of complexly shaped bodies in multibody dynamics has been presented. The surfaces of the bodies are represented by polygon meshes which are a widely–used standard in VR. For determination of the contact forces the contact patches are discretised by a geometrical algorithm closely related to computer graphics. Subsequently an extended version of the simple elastic foundation model and a regularised version of Coulomb's friction law are applied to the contact elements.

In simulations of complex test models the algorithm proved to work robustly and efficiently. Critical configurations like multiple and multiply bordered contact patches and conforming contacts were handled reliably without the problems of single contact point methods.

High requirements to the quality of the polygonal surfaces and force law parameters lacking in practical relevance appear disadvantageous. The latter problem could be eliminated by implementing an alternative contact model which is also based on an areal discretisation of the contact patch, e.g. the well investigated and powerful half–space approximation.

In the near future the numerical problems observed in the nearly static phase of the pedestrian crash simulation are going to be investigated. Besides a quantitative verification of the method is still to come.

Since the source code of the algorithm is public domain the research community can access the current state of the implementation.

References

[1] B. Simeon, C. Führer, P. Rentrop. Differential algebraic equations in vehicle system dynamics. *Survey on Mathematics in Industry*, 1:1–37,1991.

[2] F. Pfeiffer, P. Wolfsteiner. Relative kinematics of multibody contacts. In *Proceedings of the International Mechanical Engineering Congress and Exposition*. American Society of Mechanical Engineers, Dallas, Texas, 1997.

[3] F. Pfeiffer, C. Glocker. *Multibody dynamics with unilateral contacts*. Wiley&Sons New York, 1996.

[4] P. Eberhard. *Kontaktuntersuchungen durch hybride Mehrkörpersystem / Finite Elemente Simulationen*. Habilitation, Shaker Verlag Aachen, 2000.

[5] A. Joukhadar, A. Wabbi, C. Laugier. Fast contact localisation between deformable polyhedra in motion. In *Proceedings of the IEEE computer animation conference*, Geneve, 1996.

[6] S. Goyal, E.N. Pinson, F.W. Sinden. Simulation of dynamics of interacting rigid bodies including friction I: General problem and contact model. *Engineering with computers*, 10:162–174, 1994.

[7] S. Goyal, E.N. Pinson, F.W. Sinden. Simulation of dynamics of interacting rigid bodies including friction II: Software system design and implementation. *Engineering with computers*, 10:175–195, 1994.

[8] C.A. Tenaglia, D.E. Orin, R.A. LaFarge, C. Lewis. Toward development of a generalized contact algorithm for polyhedral objects. In *Proceedings of the IEEE international conference on robotics and automation*, Detroit, Michigan, 1999.

[9] *Wavefront object file specification*: http://dev.galador.net/GLViewer/obj_spec.pdf

[10] G. Hippmann. Contact mechanics in multibody systems. ECMI–workshop on numerical methods in multibody dynamics, Bad Herrenalb, 2001.

[11] K.L. Johnson. *Contact mechanics*. Cambridge University Press, 1985.

[12] J.J. Kalker. *Three–dimensional elastic bodies in rolling contact*. Kluwer Academic Publishers, 1990.

[13] J.J. Kalker, Y. van Randen. A minimum principle for frictionless elastic contact with application to non–Hertzian half–space contact problems. *Journal of engineering mathematics*, 6(2):193–206, 1972.

[14] G. Zachmann. *Virtual reality in assembly simulation – Collision detection, simulation algorithms and interaction techniques*. PhD thesis, Department of Computer Science, Darmstadt University of Technology, 2000.

[15] G. Zachmann. Rapid collision detection by dynamically aligned DOP–trees. In *Proceedings of IEEE Virtual Reality Annual International Symposium (VRAIS)*. Atlanta, Georgia, 1998.

[16] T. Möller. A fast triangle–triangle intersection test. *Journal of graphics tools*, 2(2):25–30, 1997. http://www.acm.org/jgt/papers/Moller97

[17] G. Zachmann. *Exact and fast collision detection*. Diploma thesis, Department of Computer Science, Darmstadt University of Technology, 1994.

[18] K.E. Brenan, S.L. Campbell, L.R. Petzold. *Numerical solution of initial–value problems in differential–algebraic equations*. Society for Industrial and Applied Mathematics, Philadelphia, 1996.

[19] A.C. Hindmarsh. LSODE and LSODI, two new initial value ordinary differential equation solvers. *ACM–SIGNUM Newsletters*, 15:10–11, 1980.

[20] A. Sporrer. *Grundlagen und Modellierung eines biomechanischen Mehrkörpersystems des Menschen zur Computersimulation von Bewegungsabläufen bei rechtsmedizinischen Fragestellungen*. PhD thesis, Institute of Forensic Medicine, University of Munich, 2000.

[21] D.A. Winter. *Biomechanics and motor control of human movement*. John Wiley & Sons Inc., 1990.

[22] *Meta3D website*: http://www.exchange3d.com

[23] *3D Cafe website*: http://www.3dcafe.com

[24] *AC3D v3.6 user manual*, Inivis Ltd., Ely, Cambs, United Kingdom, 2003. http://www.ac3d.org

[25] E. Hairer, S.P. Nørsett, G. Wanner. *Solving ordinary differential equations I: Nonstiff problems*. Second revised edition, Springer–Verlag, 1993.

[26] *PCM website*: http://www.pcm.hippmann.org