Università degli Studi di Roma Tre
Dipartimento di Informatica e Automazione
Via della Vasca Navale, 79 – 00146 Roma, Italy

A unified multidimensional approach to extrusion, sweeping, offset and Minkowski sum

Alberto Paoluzzi

RT-DIA-30-97 **Dicembre 1997**

Università di Roma Tre,
Via della Vasca Navale, 79
00146 Roma, Italy.
Email: `paoluzzi@inf.uniroma3.it`

ABSTRACT

AAA

1 Introduction

A multidimensional approach which unifies extrusion, sweeping and a subclass of offset and Minkowski sums is given in this paper. This approach is given for linear, non regular and non manifold d-polyhedra embedded in a Euclidean n-space.

The addressed operations are useful in several areas of CAD/CAM and robotics. Extrusion is used to generate 3D models from 2D sections. Sweeping is used to generate curves, surfaces and solids by moving points, curves, surfaces or solids along given paths. Minkowski sum is referred to when planning collision-free motion of vehicles among obstacles. The subclass of offset operations studied here allows one to automatically generate solid models of buildings from polygon and wire-frame models.

To this purpose a construction is introduced, which is based on the repeated execution of the Cartesian product of a point-set in E^n times a unit interval, followed by a proper shearing. The partial or final results of a suitable sequence of such intermediate operations are projected back into E^n.

In particular, *extrusion* is here defined as the Cartesian product of a polyhedral point-set A times the unit interval $[0, 1] \subset E$. *Sweeping* is defined as the union of joins of corresponding points in a polyhedral point-set and in its translated image. An alternative definition of sweeping is introduced as the Minkowski sum with an interval. It is shown that sweeping can be computed as a composition of extrusion, shearing and projection. If a convex n-cell K is computable as a succession of sweeps from a single point, then the *Minkowski sum* of any d-polyhedron A with K can be computed by applying the same sweeps to A. An *orthogonal offset* operator is defined as the Minkowski sum of a polyhedral point-set with a hyper-rectangle. A constructive definition of offset is given as the composition of suitable sweeps.

The present approach is limited to polyhedral domains, but it can be used with polyhedral complexes of any dimension d, with $0 < d < n$ (for any n, even bigger than 3). Such point-sets may be both solid and embedded, non regular and non manifold.

The operations considered and some of their parental relationships are not new. For example, the Minkowski sum of 2D polygons is referred by Schwartz and Sharir [7] to the early work of Lozano-Perez and Wesley [4]. The novelty of the present approach relies on its close integration in a multidimensional computing environment. Among the ideas which made this approach possible: the representation of a point-set by a *covering* with convex cells [2, 3, 5]; the representation of a convex set as intersection of halfspaces [1], *no representation of topology* [8], a *multidimensional* approach to both data structures and algorithms, and a *functional* geometric environment [5]. No topology representation implies simplicity, efficiency and robustness, beyond easy dimension-independence.

2 Background

According to the standard usage in modeling and graphics, normalized homogeneous coordinates $(x_0, x_1, \dots x_n)$ are used, where $x_0 = 1$, to represent any point $p \in E^n$.

Convex set A non-empty subset K of E^n is said to be *convex* if $\alpha x + (1-a)y$ is in K for whatever $x, y \in K$ and $0 \leq \alpha \leq 1$. The convex set $[x, y]$ is called the *line segment* joining x and y. Each convex set can be generated as a finite intersection of halfspaces. The *dimension* d of a convex set K is the dimension of the smallest flat containing K.

Polyhedron Polyhedra can be represented as coverings with convex sets. The polyhedra we deal with are non-convex, non-regular (i.e. may have parts of different dimension), non-connected and non-manifold. Any polyhedral point-set $A \subset E^n$ can be written as $\bigcup_i K_i$, where $\{K_i\}$ is a collection of relatively closed convex sets in E^n, called *cells*. The dimension d of a polyhedron is the maximum dimension of its convex cells. The set of polyhedra of dimension d in E^n is denoted as $\mathcal{P}^{d,n}$.

Product of polyhedra Polyhedral sets can be multiplied by pairwise cartesian product of their cells. If $A_1 = \bigcup_i K_i$ and $A_2 = \bigcup_j K_j$, with $K_i \subset E^n$ and $K_j \subset E^m$, then

$$A_1 \times A_2 = \bigcup_{i,j} (K_i \times K_j), \quad \text{where} \quad A_1 \times A_2 \subset E^{n+m}.$$

Extrusion The unary *extrusion* operation is defined as the Cartesian product of a point-set times the unit interval $[0,1] \subset E$:

$$\mathrm{Extr} : E^n \to E^{n+1} :$$

$$A = \bigcup_i K_i \mapsto \mathrm{Extr}(A) = \bigcup_i (K_i \times [0,1]).$$

Shearing The *shearing* is defined as a special linear transformation H of E^n. The matrix representing such a transformation in a suitable basis differs from the unit matrix only for one column.

A shearing transformation produces a special deformation of the space which is applied to, considered as a bundle of hyperplanes parallel to a coordinate subspace of dimension $n-1$. Under the shearing such coordinate subspace (let $x_n = 0$) is fixed, whereas all the parallel hyperplanes are translated linearly with x_n.

A shearing transformation along the x_n coordinate is represented as a $(n+1) \times (n+1)$ matrix with the following structure:

$$H_v = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & v_1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & v_{n-1} \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix}$$

The matrix depends on a vector $v = (v_1, \ldots, v_{n-1}) \in \Re^{n-1}$, which represents the translation acting on the points of the hyperplane $x_n = 1$. The application of the shearing H_v to the point-set A is denoted as $H_v(A)$.

Projection A *projection* Π^r is here defined as a linear operator $\Pi^r : E^n \to E^{n-r}$ which maps a polyhedron $A = \bigcup_i K_i$, into a polyhedron $\Pi^r(A) = \bigcup_i \Pi^r(K_i)$, where each $\Pi^r(K_i)$ is the convex hull of the projected vertices of K_i.

3 Unified approach

The Minkowski sum is usually defined [6, 10] between convex sets in a vector space, so producing a location of the result which is different from that of both arguments. The definition in a Euclidean space given here follows [4] and [7], and produces a "grown" version of the first argument.

Definition 1 (Minkowski sum) *Consider a polyhedron $A = \bigcup A_i$, where A_i are convex cells, and another* convex *K. Choose a point $o \in K$ as the origin for a support vector space $\Re^n$. Let $x = a - o$ and $y = k - o$ be the vectors spanning A_i and K, respectively. For each convex cell A_i, consider the set*

$$A_i + K = \{o + (a - k) \mid a \in A_i,\ k \in K\},$$

where $a - k = x - y$ is the vector (Minkowski) difference of the spanning vectors x and y. The polyhedron obtained as the union of the expanded sets $A_i + K$ is called the Minkowski sum *of A with the convex K:*

$$A + K = (\cup_i A_i) + K = \cup_i (A_i + K)$$

Notice that polyhedra are defined as point-sets in an Euclidean space E^n, while the difference of two points gives a vector in the underlying vector space, and the sum of a point and a vector gives a point.

In the following, it is supposed that an origin has been chosen once for all, in order to identify the Euclidean space and the associated vector space.

Definition 2 (Sweeping) *The* sweeping *operation may be defined as the Minkowski sum of a polyhedral point-set A and the interval $[0, v]$*

$$Sweep(v) : E^n \to E^n : A \mapsto A + [0, v].$$

If the set of vectors $\{v_1, v_2, \dots v_k\}$ is linearly independent, then the set $\sum_{i=1}^{k} [0, v_i]$ is called the *parallelotope* spanned by those vectors.

Let denote $[-v_i/2, v_i/2]$ as $[v_i]$. If $V_m = \{v_1, v_2, \dots v_m\}$ is any set of m vectors, then the set $[V_m] = \sum_{V_m} [v_i]$ is called the *zonotope* spanned by the v_i. Any zonotope is symmetric, i.e., if $w \in [V_m]$, then $-w \in [V_m]$. This justifies the following definition.

Definition 3 (Offset) *The* Offset *operator can be defined as the Minkowski sum of a polyhedral point-set A with a zonotope $[V_n]$.*

A special interesting case, with applications to building design, is when V_n constitutes an orthogonal basis. By the associative property of Minkowski sum a constructive definition of the offset operator can be given as the composition of $2n$ suitable sweeps:

$$A + [V_m] = (\mathrm{Sweep}(-v_m/2) \circ \mathrm{Sweep}(v_m/2) \circ \cdots \circ \mathrm{Sweep}(-v_1/2) \circ \mathrm{Sweep}(v_1/2))(A)$$

4 Algorithms

The Minkowski sum with an interval, i.e. a sweeping in E^n, is reduced to computing a product times the unit interval, to an affine transformation in E^{n+1} and to a projection from E^{n+1} to E^n. The Minkowski sum with either a parallelotope or a zonotope generated by m vectors is reduced to a succession of m products (each associated with an affine transformation) and finally to a single projection from E^{n+m} to E^n.

Proposition: The *Minkowski sum with an interval* $[0, v]$ can be computed as the composition of extrusion, shearing and projection operators:

$$\mathrm{A} + [0, v] = (\Pi \circ H_v \circ \mathrm{Extr})(A) \tag{1}$$

Proposition: A succession of m sweeps is equivalent to the alternate composition of m extrusions and shearings, followed by a projection Π^m:

$$\mathrm{Sweep}(v_n) \circ \cdots \circ \mathrm{Sweep}(v_1) = \Pi^n \circ (H_{s_n} \circ \mathrm{Extr}) \circ \cdots \circ (H_{s_1} \circ \mathrm{Extr})$$

where the shearing vectors s_i are derived by the sweeping vectors v_i by adding one more (zero) coordinate at a time:

$$s_1 = v_1, \quad s_{i+k} = v_i \times \{0\}^k$$

Proposition: If a convex cell K is computable by sweeps from a single point, i.e. if

$$K = (\mathrm{Sweep}(r_m) \circ \cdots \circ \mathrm{Sweep}(r_1))(p), \qquad p \in E^n,$$

then the Minkowski sum $A + K$ of any polyhedron A with K can be computed by applying the same sweeps to A. More formally:

$$A + K = (\mathrm{Sweep}(r_m) \circ \cdots \circ \mathrm{Sweep}(r_1))(A)$$

5 Examples

Example 5.1 *(Extrusion and shearing)*

Let $A = \partial I^2$*, with* $I^2 = [0,1] \times [0,1] \subset E^2$*. Notice that* $\partial I^2 \in \mathcal{P}^{1,2}$*. The result of the application of a shearing* $H_v = H(0.1, 0.2)$ *to the point-set* $Extr(A) = (\partial I^2) \times I$ *is shown in Figure 1.*

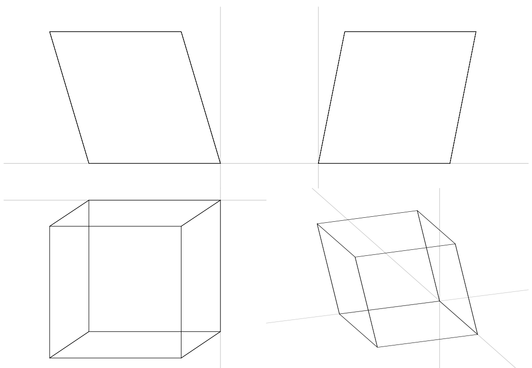

Figure 1: Extruded and sheared object $H(0.3, 0.2)(\partial I^2 \times I)$.

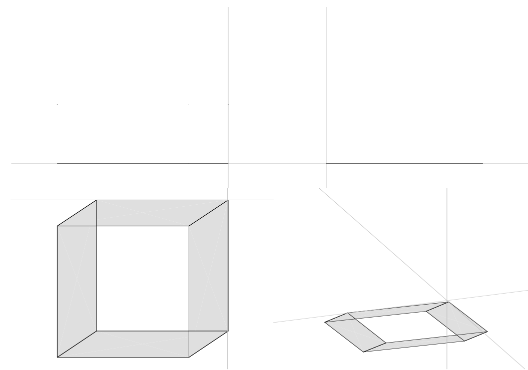

Figure 2: Covering with convex cells of the point-set $\mathrm{Sweep}(0.3, 0.2)(\partial I^2)$.

Example 5.2 *(Sweeping)*

A simple example of sweeping can be given by applying the composite operator shown in Formula (1) to the point-set $A = \partial I^2$*. Let the translation vector* t *be equal to* $(0.3, 0.2)$*. The swept object* $Sweep_t(A)$*, computed as* $(i \circ H(t) \circ Extr)(A)$ *is shown in Figure 2.*

Example 5.3 *(Multiple sweeping)*

A more interesting example can be given by first generating a 2D polyhedral complex embedded in 3D and then by repeatedly sweeping it in 3D. Let suppose that

$$B = A \cup T(A),$$

where $A = \partial I^2 \times [0, 0.3]$ *and* T *is a translation of* E^3 *with translation vector* $t = (1,0,0)$*. Let compute the 3D object defined as:*

$$(Sweep(v_3) \circ Sweep(v_2) \circ Sweep(v_1))(B)$$

where $v_1 = (-0.1, 0.2, 0)$, $v_2 = (0.1, 0, 0)$ *and* $v_3 = (0, 0.2, 0.3)$. *Both the partial and the final results of the computation performed using Formula (1) are shown in Figure 3.*

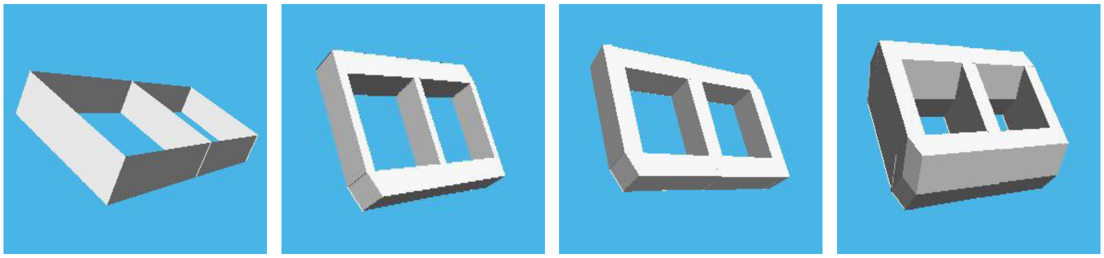

Figure 3: (a) Polyhedral 2-complex $B = A \cup T(A)$. (b) Polyhedral 3-complex $C = \text{Sweep}(-0.1, 0.2, 0)(B)$. (c) Polyhedral 3-complex $D = \text{Sweep}(0.1, 0, 0)(C)$. (d) Polyhedral 3-complex $E = \text{Sweep}(0, 0.2, 0.3)(D)$.

Example 5.4 *(Offset) Let* σ^2 *be the 2-dimensional simplex, i.e. the convex combination of points* $(0, 0)$, $(0, 1)$, *and* $(1, 0)$. *So we have* $\sigma^2 \in \mathcal{P}^{(2,2)}$ *and* $\partial\sigma^2 \in \mathcal{P}^{(1,2)}$. *In Figure 4 we show Offset(*r*)(*$\partial\sigma^2$*) and Offset(*r*)(*σ^2*), derived by sweeping* $\partial\sigma^2$ *and* σ^2 *with sweeping vector* $r = (0.2, 0.2)$, *respectively.*

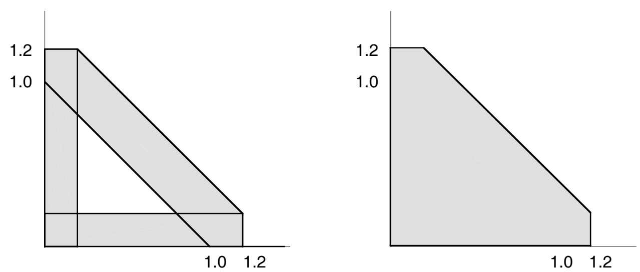

Figure 4: Offset objects $\text{Offset}(0.2, 0.2)(\partial\sigma^2)$ and $\text{Offset}(0.2, 0.2)(\sigma^2)$.

Example 5.5 *(Offset) A more interesting example of offset is given by developing a 2-complex in* E^3, *which models the vertical walls of a simple house, and by offsetting it. Both the starting complex and the offsetted 3D result are shown in Figure 5.*

Example 5.6 *(Offset with orthogonal parallelotope) A Minkowski sum with the cuboid of extreme point* $(0.1, 0.2, 0.1)$ *is given starting from the* 1*-complex* B *in* E^3 *shown in Figure 6a. The "solid" result is in Figure 6b.*

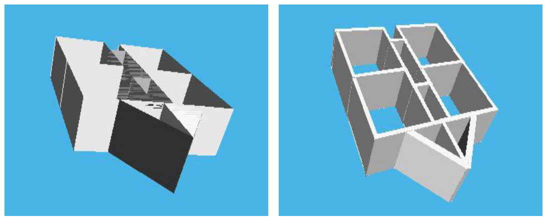

Figure 5: (a) 2-complex A in E^3. (b) 3-complex Offset$(0.2, 0.2)(A)$.

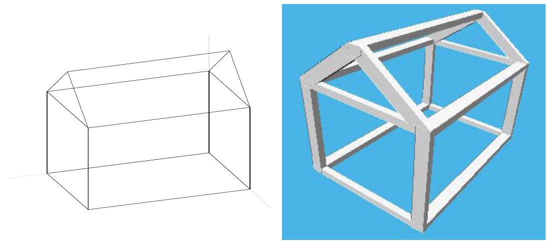

Figure 6: (a) 1-complex B in E^3. (b) 3-complex Offset$(0.1, 0.2, 0.1)(B)$.

Example 5.7 *(Offset)*

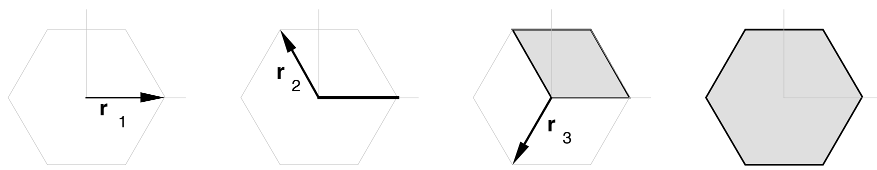

Figure 7: Tre three sweeping vectors which generate the hexagonal 2-cell shown in the last picture.

In this example we generate a convex 2-cell B and then apply a Minkowski sum with it both to a 2-complex A and to a 1-complex ∂A. The convex B is generated starting from $(0,0)$ as follows. The generation process is graphically shown in Figure 7.

$$B = \left(Sweep\left(-\frac{1}{2}, -\frac{\sqrt{3}}{2}\right) \circ Sweep\left(-\frac{1}{2}, \frac{\sqrt{3}}{2}\right) \circ Sweep(1,0) \right) (0,0)$$

A polyhedral 2-complex A is then defined as $I^2 - K^2$, where the minus denotes a Boolean subtraction, and $I = [0,1]$, $K = [0.5, 1]$. The results of computing the expressions $A + B$ and $\partial A + B$ with this approach are shown in Figure 8. In Figure 9 an assembly of extruded instances of the grown complexes corresponding respectively to the 2-complex A and to its 0-skeleton and boundary is shown.

6 Analysis

This approach is efficient. If the convex cells are represented as the collection of a bounded number of intersecting halfspaces, then the Cartesian product [1] times an interval is $O(m)$ in time and space, where m is the number of convex cells in

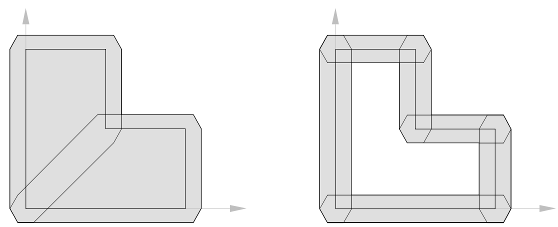

Figure 8: (a) The result of the operation $A + B$, with $A = I^2 - K^2$. Notice that there are two convex cells in A. (b) The result of $\partial A + B$.

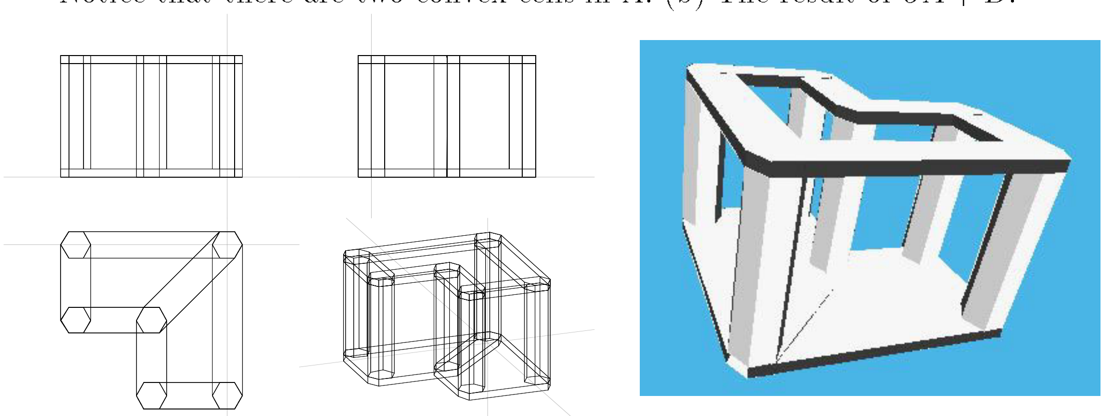

Figure 9: (a) Several wire-frame projections of the assembly of extruded complexes obtained as Minkowski sums for an hexagonal cell. (b) VRML image of the same assembly.

the input. In fact if a cell has representation of bounded size n, after the product by an interval its representation has size $n+2$ and is bounded also.

The shearing is linear in the size of the representation. The projection of convex cells is more costly, but the size of cells is bounded. Actually the cost is that of computing the convex hull of projected cell vertices. This may imply the computation of vertices starting from face inequalities, and the number of vertices grows exponentially with the number of extrusions. A better approach is to directly project the face inequalities using the Fourier-Motzkin algorithm [9].

As a consequence, each linear sweep is $O(cm)$, where c is the cost of the projection of the bounded cell, so that the $A+K$ Minkowski sum in E^n is $O(cpm)$, where m is the cell number of A and p is the number of sweeps needed to generate the convex K.

Acknowledgments The author would like to thank Antonio di Carlo and Valerio Pascucci for comments on a preliminary version of the manuscript and for providing bibliography on Minkowski geometry, and Chandrajit Bajaj for friendly encouragement.

References

[1] Bernardini, F., Ferrucci, V., Paoluzzi, A. and Pascucci, V. "A Product Operator on Cell Complexes". *Proc. of the ACM/IEEE 2nd Conf. on Solid Modeling and Appl.*, ACM Press, 43–52, February 1993.

[2] Günther, O., and Wong, E. "Convex polyhedral chains: A representation for geometric data". *Computer Aided Design*, 21(3), March 1989, 157–164.

[3] Hedelsbrunner, H. "Algebraic decomposition of non-convex polyhedra". In *36th Symp. on Found. of Comp.Sc.*, 248–257. IEEE, October 1995.

[4] Lozano-Perez, T. and Wesley, M. "An Algorithm for planning collision-free paths among polyhedral obstacles". *Communications of the ACM*, **22**(3): 560–570, 1979.

[5] Paoluzzi, A., Pascucci, V., and Vicentino, M. “Geometric programming: a programming approach to geometric design”. *ACM Trans. on Graphics*, **14**(4):266–306, July 1995.

[6] Schneider, R. *Convex Bodies: The Brunn-Minkowski Theory.* Encyclopedia of Mathematics and its Applications, v.44, Cambridge Univ. Press, 1993.

[7] Schwartz, J. T. and Sharir, M. “Algorithmic Motion Planning in Robotics”. in *Handbook of Theoretical Computer Science, Vol. A, Algorithms and Complexity*, J. Van Leeuwen (Ed.), Elsevier, Amsterdam, 1990.

[8] Takala, T. “Geometric Boundary Modelling Without Topological Data Structures”. In *Eurographics ‘86*, A.A.G. Requicha, Ed., North Holland, 1986.

[9] Thapa, M. N. and Dantzig G.B. *Linear Programming 1: Introduction.* Springer Series in Operation Research, Springer-Verlag, 1997.

[10] Thompson, A. C. *Minkowski Geometry.* Encyclopedia of Mathematics and its Applications, v.63, Cambridge Univ. Press, 1996.

APPENDIX. Implementation

The *complete* implementation of the approach discussed in this paper is given here using the `PLaSM` geometric language [5]. The `OffSet` operator is here implemented as a sum with an orthogonal parallelotope with extreme vertices 0 and v.

First some utility functions are given, followed by the implementation of the discussed operators. Primitive functions in the language are written with capital letters.

The interested reader is referred to the web pages at URL
`http://monge.inf.uniroma3.it/Plasm/plasm0.html`

```
DEF IsVect = ISSEQOF:ISREAL;
DEF IsMat = AND ~ [ISSEQOF:IsVect, EQ ~ AA:LEN];
DEF IsSqrMat =  AND ~ [IsMat, EQ ~ [LEN, LEN ~ s1]];
DEF vet2mat = (AA~AA):* ~ AA:TRANS ~ DISTR ~ [IDNT~LEN, ID];
DEF Scaling (s::ISREAL) (v::IsVect) = (AA:* ~ DISTL):<s,v>;
DEF Q = QUOTE ~ IF:<IsSeq,ID,LIST>;
DEF MatHom (m::IsSqrMat) = AL:< firstRow, (AA:AL ~ DISTL):<0,m> >
    WHERE
```

```
        firstRow = AL:<1,#:(LEN:m):0>
    END;
DEF IDNT (n::ISINTPOS) = (AA ~ AA):Kroneker:(CartProd:<1..n,1..n>)
    WHERE
        Kroneker = IF:<EQ, K:1, K:0>,
        CartProd = AA:DISTL ~ DISTR
    END;
DEF ExtrAndShear (v::IsVect) (pol::ISPOL) = Shear:v:(pol * QUOTE:<1>);
DEF Shear (v::IsVect) =
        (MAT ~ MatHom ~ TRANS ~ AR ~ [ButLast~IDNT~LEN, ID] ~ AR):<v,1>
    WHERE
        ButLast = Reverse~TAIL~Reverse
    END;
DEF MultipleExtr (v::IsVect)  =
        (COMP ~ AA:ExtrAndShear ~ REVERSE ~ AppendZeros ~ vet2mat):v;
DEF AppendZeros (m::IsSqrMat) = (AA:CAT ~ TRANS ~ [ID, GenerateZeroSeqs]): m
    WHERE
        GenerateZeroSeqs =  AL ~ [ K:<>, APPLY~[
            CONS ~ AA:# ~ FROMTO ~ [K:1, -~[LEN,K:1]],  K:0 ] ]
    END;
DEF Project (n::ISINT)(pol::ISPOL) =
        (MKPOL ~ [AA:CutCoords~S1, S2, S3] ~ UKPOL):pol
    WHERE
        CutCoords = Reverse~(COMP:(#:n:TAIL))~Reverse
    END;

DEF Sweep  (v::IsVect)(pol::ISPOL) = (Project:1 ~ ExtrAndShear:v):pol;
DEF OffSet (v::IsVect)(pol::ISPOL) = (Project:(LEN:v) ~ MultipleExtr:v):pol;
DEF Minkowsky (vects::ISSEQOF:IsVect)(p::ISPOL) = (COMP ~ AA:Sweep):vects:p;
```