Parallel & Distibuted Computing: Lecture 3

Alberto Paoluzzi

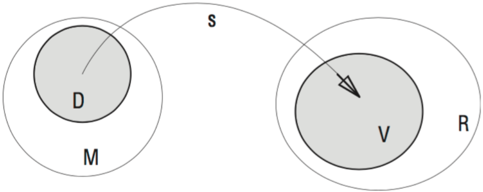
March 14, 2017

Introduction to geometric modeling

Solid modeling

Representation scheme

mapping $s : M \to R$ from a space of math models M to computer representations R



1. The set M contains the mathematical models of the class of solid objects that the scheme aims to represent.
2. The set R contains symbolic representations, i.e. suitable data structures, built according to some appropriate computer grammar.

Some types of representation schemes

- Primitive instancing
- Quasidisjoint decomposition
- (Simple) sweeping
- Boundary representation
- Constructive solid geometry (CSG)

Volume integration in CAD/CAE

Volume, moments of inertia, and similar properties of solids are defined by triple (volumetric) integrals over subsets of 3D Euclidean space.

The automatic computation of integral properties for geometrically complex solids is important in CAD/CAM/CAE (in a single word: PLM), and robotics

(Lee and Requicha 1982) discusses methods for computation of

$$\int_S f(P)dV$$

where $P = (x, y, z) \in \mathbb{E}^3$), dV is the volume differential, f is a real-valued scalar function, e.g., a polynomial, and S is a solid (r-set) that may be geometrically complex, e.g., a typical mechanical part bounded by many curved faces.

Typical integrals in mechanical CAD

$$\int_S f(P) dV$$

$f(P) = p$

$I = m$, or volume,when $p = 1$

$f(P) = x/m$

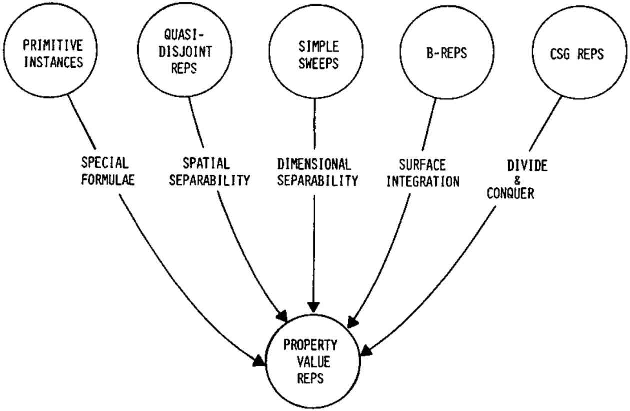
$I = x$-coordinate of barycenter

$f(p) = x^2 + y^2$

$I =$ moment of inertia about z

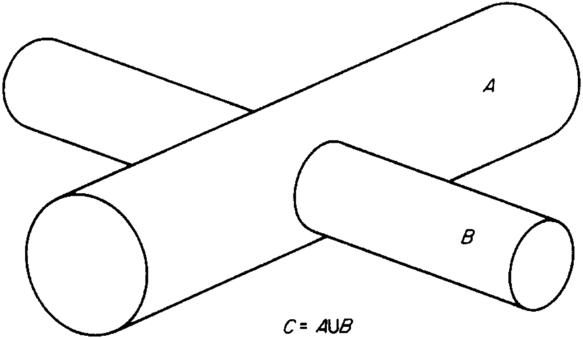
Integration in Solid Modeling

"Natural" integration methods



Timmer-Stern Method for B-reps w parametric surfaces

See "Computation of global geometric properties of solid objects"



Symbolic solution to domain integration of polynomials

Problem formulation

Finite evaluation of integrals:

$$II_S \equiv \iint_S f(\mathbf{p})\, dS,$$

$$III_P \equiv \iiint_P f(\mathbf{p})\, dV,$$

where S, and P are linear and regular 2- or 3-polyhedra in $\mathbb{R}^3$, dS and dV are the differential surface and the differential volume.

Problem formulation

Finite evaluation of integrals:

$$II_S \equiv \iint_S f(\mathbf{p})\, dS,$$

$$III_P \equiv \iiint_P f(\mathbf{p})\, dV,$$

where S, and P are linear and regular 2- or 3-polyhedra in $\mathbb{R}^3$, dS and dV are the differential surface and the differential volume.

The integrating function is a trivariate polynomial

$$f(\mathbf{p}) = \sum_{\alpha=0}^{n} \sum_{\beta=0}^{m} \sum_{\gamma=0}^{p} a_{\alpha\beta\gamma} x^{\alpha} y^{\beta} z^{\gamma},$$

where α, β, γ are non-negative integers.

Problem solution

See "Boundary integration over linear polyhedra"

In the following we summarize from (Cattani and Paoluzzi 1990) an "exact" symbolic solution both to the surface and volume integration of polynomials, by using a triangulation of the volume boundary.

The evaluation of volume integrals is achieved by transforming them into line integrals over the boundary of every simplex of a boundary triangulation of integration domain

Finite integration over polyhedra

Problem statement The finite method [CP90] to compute double and triplet integrals of monomials over linear regular polyhedra in $\mathbb{R}^3$ is discussed. In particular, this method enables practical formulae for the exact evaluation of integrals to be achieved:

$$II_S \equiv \int\!\!\int_S f(\mathbf{p})\, dS, \qquad III_P \equiv \int\!\!\int\!\!\int_P f(\mathbf{p})\, dV, \tag{1}$$

where S, and P are linear and regular 2- or 3-polyhedra in $\mathbb{R}^3$, dS and dV are the differential surface and the differential volume. The integrating function is a trivariate polynomial

$$f(\mathbf{p}) = \sum_{\alpha=0}^{n} \sum_{\beta=0}^{m} \sum_{\gamma=0}^{p} a_{\alpha\beta\gamma} x^\alpha y^\beta z^\gamma,$$

where α, β, γ are non-negative integers.

Since the extension to $f(\mathbf{p})$ is straightforwardly given by the linearity of integral operator, we may focus on the calculation of integrals of monomials:

$$II_S^{\alpha\beta\gamma} \equiv \int\!\!\int_S x^\alpha y^\beta z^\gamma\, dS, \qquad III_P^{\alpha\beta\gamma} \equiv \int\!\!\int\!\!\int_P x^\alpha y^\beta z^\gamma\, dV. \tag{2}$$

Finite integration over polyhedra

Surface integration We call *structure product* the integral of a monomial over a simplicial complex. Exact formulae for structure products over n-sided polygons in 2-space, the unit triangle in 2-space, and an arbitrary triangle in 3-space, are derived in the following. Structure products are a generalization of the usual products and moments of inertia, that can be obtained from (2) by assuming $\alpha + \beta + \gamma \leq 2$.

Finite integration over polyhedra

Polygon integrals A structure product over a polygon π in the plane xy is

$$II_\pi^{\alpha\beta} = \iint_\pi x^\alpha y^\beta \, dS, \qquad \alpha, \beta \geq 0, integers. \tag{3}$$

Such integrals can be exactly expressed, when π is a polygon with n oriented edges, as:

$$II_\pi^{\alpha\beta} = \frac{1}{\alpha+1} \sum_{i=1}^{n} \sum_{h=0}^{\alpha+1} \binom{\alpha+1}{h} x_i^{\alpha+1-h} X_i^h \sum_{k=0}^{\beta} \frac{\binom{\beta}{k}}{h+k+1} y_i^{\beta-k} Y_i^{k+1} \tag{4}$$

where $\mathbf{p}_i = (x_i, y_i)$, $X_i = x_{i+1} - x_i$, $Y_i = y_{i+1} - y_i$ and $\mathbf{p}_{n+1} = \mathbf{p}_1$. The derivation of the formula (4) is based on the application of Green's theorem and on Newton's expression for binomial powers.

Finite integration over polyhedra

Unit triangle integrals The general formula (4) can be specialized for the unit triangle $\tau' = \langle \mathbf{w}_o, \mathbf{w}_a, \mathbf{w}_b \rangle$, with vertices

$$\mathbf{w}_o = (0,0), \qquad \mathbf{w}_a = (1,0), \qquad \mathbf{w}_b = (0,1), \tag{5}$$

getting a very simplified expression. With some algebraic manipulations, we obtain[1]

$$II^{\alpha\beta} = \frac{1}{\alpha+1} \sum_{h=0}^{\alpha+1} \binom{\alpha+1}{h} \frac{(-1)^h}{h+\beta+1}, \tag{6}$$

which reduces, for $\alpha = \beta = 0$, to the area of the triangle (5): $II^{00} = 1/2$.

[1] $II_\pi^{\alpha\beta}$ is substituted, when referred to the unit triangle, by the symbol $II^{\alpha\beta}$.

Finite integration over polyhedra

Triangle integrals In the following we derive the general expression for structure products evaluated on an arbitrary triangle $\tau = \langle \mathbf{v}_o, \mathbf{v}_a, \mathbf{v}_b \rangle$ of the 3-space xyz, defined by $\mathbf{v}_o = (x_o, y_o, z_o)$ and by the vectors $\mathbf{a} = \mathbf{v}_a - \mathbf{v}_o$ and $\mathbf{b} = \mathbf{v}_b - \mathbf{v}_o$. The parametric equation of its embedding plane is:

$$\mathbf{p} = \mathbf{v}_o + u\,\mathbf{a} + v\,\mathbf{b}, \tag{7}$$

where the area element is

$$d\tau = |J|\,du\,dv = \left|\frac{\partial \mathbf{p}}{\partial u} \times \frac{\partial \mathbf{p}}{\partial v}\right| du\,dv = |\mathbf{a} \times \mathbf{b}|\,du\,dv. \tag{8}$$

A structure product over a triangle τ in 3-space can be transformed by a coordinates transformation, as follows:

$$II_\tau^{\alpha\beta\gamma} = \iint_\tau x^\alpha y^\beta z^\gamma\,d\tau = |\mathbf{a} \times \mathbf{b}| \iint_{\tau'} x^\alpha(u,v) y^\beta(u,v) z^\gamma(u,v)\,du\,dv, \tag{9}$$

where τ' is the uv domain that corresponds to τ under the coordinate transformation (7).

Finite integration over polyhedra

In this case we have (the proof is given in [?]):

$$\begin{aligned} II_\tau^{\alpha\beta\gamma} &= |\mathbf{a}\times\mathbf{b}| \sum_{h=0}^{\alpha} \binom{\alpha}{h} x_o^{\alpha-h} \sum_{k=0}^{\beta} \binom{\beta}{k} y_o^{\beta-k} \sum_{m=0}^{\gamma} \binom{\gamma}{m} z_o^{\gamma-m} \cdot \\ &\cdot \sum_{i=0}^{h} \binom{h}{i} a_x^{h-i} b_x^i \sum_{j=0}^{k} \binom{k}{j} a_y^{k-j} b_y^j \sum_{l=0}^{m} \binom{m}{l} a_z^{m-l} b_z^l \, II^{\mu\nu}, \end{aligned} \tag{10}$$

where $\mu = (h+k+m) - (i+j+l), \nu = (i+j+l)$, and $II^{\mu\nu}$ is a structure product over the triangle (5), as given by formula (6). Of course the area of a triangle τ is:

$$II_\tau^{000} = \iint_\tau d\tau = |\mathbf{a}\times\mathbf{b}| \, II^{00} = \frac{|\mathbf{a}\times\mathbf{b}|}{2}. \tag{11}$$

Surface integrals In conclusion, a structure product over a polyhedral surface S, open or closed, is a summation of structure products (10) over the 2-simplices of a triangulation K_2 of S:

$$II_S^{\alpha\beta\gamma} = \iint_S x^\alpha y^\beta z^\gamma \, dS = \sum_{\tau\in K_2} II_\tau^{\alpha\beta\gamma}. \tag{12}$$

Finite integration over polyhedra

Volume integration Let P be a three-dimensional polyhedron bounded by a polyhedral surface ∂P. The regularity of the integration domain and the continuity of the integrating function enable us to apply the divergence theorem, which can be briefly summarized, for a vector field $\mathbf{F} = \mathbf{F}(\mathbf{p})$ as:

$$\iiint_P \nabla \cdot \mathbf{F}\, dx\, dy\, dz = \iint_{\partial P} \mathbf{F} \cdot \mathbf{n}\, dS = \sum_{\tau \in K_2} \iint_\tau \mathbf{F} \cdot \mathbf{n}_\tau\, d\tau, \tag{13}$$

where $\mathbf{n}$ is the outward vector normal to the surface portion dS, and hence $\mathbf{n}_\tau = \mathbf{a} \times \mathbf{b}/|\mathbf{a} \times \mathbf{b}|$.

Finite integration over polyhedra

As the function $x^\alpha y^\beta z^\gamma$ equates the divergence of the vector field $\mathbf{F} = (x^{\alpha+1} y^\beta z^\gamma / \alpha + 1, 0, 0)$, an expression for $III_P^{\alpha\beta\gamma}$ is easily derived, which depends only on the 1-simplices of a triangulation of the domain boundary and on the structure products over its 2-simplices.

Finite integration over polyhedra

As a matter of fact, we have:

$$\begin{aligned} III_P^{\alpha\beta\gamma} &= \iiint_P x^\alpha y^\beta z^\gamma \, dx\, dy\, dz \\ &= \iiint_P \frac{\partial}{\partial x}\left(\frac{1}{\alpha+1} x^{\alpha+1} y^\beta z^\gamma\right) dx\, dy\, dz \\ &= \frac{1}{\alpha+1} \sum_{\tau' \in K'_2} (\mathbf{a}\times\mathbf{b})_x \iint_{\tau'} x^{\alpha+1} y^\beta z^\gamma \, du\, dv. \end{aligned} \tag{14}$$

Taking into account equations (8) and (9), we can substitute the integral in the previous equation, getting finally:

$$III_P^{\alpha\beta\gamma} = \frac{1}{\alpha+1} \sum_{\tau\in K_2} \left[\frac{(\mathbf{a}\times\mathbf{b})_x}{|\mathbf{a}\times\mathbf{b}|}\right]_\tau II_\tau^{\alpha+1,\beta,\gamma} \tag{15}$$

where the surface integrals are evaluated by using the formula (10).

Python & Julia implementation

Surface and volume integrals

```python
""" Surface and volume integrals """
def Surface(P, signed=False):
    return II(P, 0, 0, 0, signed)
def Volume(P):
    return III(P, 0, 0, 0)
```

Terms of the Euler tensor

```python
def FirstMoment(P):
    out = [None]*3
    out[0] = III(P, 1, 0, 0)
    out[1] = III(P, 0, 1, 0)
    out[2] = III(P, 0, 0, 1)
    return out

def SecondMoment(P):
    out = [None]*3
    out[0] = III(P, 2, 0, 0)
    out[1] = III(P, 0, 2, 0)
    out[2] = III(P, 0, 0, 2)
    return out
```

```python
def InertiaProduct(P):
    out = [None]*3
    out[0] = III(P, 0, 1, 1)
    out[1] = III(P, 1, 0, 1)
    out[2] = III(P, 1, 1, 0)
    return out
```

Vectors and covectors of mechanical interest

```
def Centroid(P):
    out = [None]*3
    firstMoment = FirstMoment(P)
    volume = Volume(P)
    out[0] = firstMoment[0]/volume
    out[1] = firstMoment[1]/volume
    out[2] = firstMoment[2]/volume
    return out

def InertiaMoment(P):
    out = [None]*3
    secondMoment = SecondMoment(P)
    out[0] = secondMoment[1] + secondMoment[2]
    out[1] = secondMoment[2] + secondMoment[0]
    out[2] = secondMoment[0] + secondMoment[1]
    return out
```

Basic integration functions

```
def M(alpha, beta):
    a = 0
    for l in range(alpha + 2):
        a += CHOOSE([alpha+1,l]) * POWER([-1,l])/(l+beta+1)
    return a/(alpha + 1)

def II(P, alpha, beta, gamma, signed):
    w = 0
    V, FV = P
    for i in range(len(FV)):
        tau = [V[v] for v in FV[i]]
        term = TT(tau, alpha, beta, gamma, signed)
        w += term
    return w
```

Basic integration functions

```
def III(P, alpha, beta, gamma):
    w = 0
    V, FV = P
    for i in range(len(FV)):
        tau = [V[v] for v in FV[i]]
        vo,va,vb = tau
        a = VECTDIFF([va,vo])
        b = VECTDIFF([vb,vo])
        c = VECTPROD([a,b])
        w += (c[0]/VECTNORM(c)) * TT(tau, alpha+1, beta, gamma)
    return w/(alpha + 1)
```

Basic integration functions

```
def TT(tau, alpha, beta, gamma, signed=False):
    vo,va,vb = tau
    a = VECTDIFF([va,vo])
    b = VECTDIFF([vb,vo])
    sl = 0;
    for h in range(alpha+1):
        for k in range(beta+1):
            for m in range(gamma+1):
                s2 = 0
                for i in range(h+1):
                    s3 = 0
                    for j in range(k+1):
                        s4 = 0
                        for l in range(m+1):
                            s4 += CHOOSE([m, l]) * POWER([a[2], m-l]) \
                                * POWER([b[2], l]) * M( h+k+m-i-j-l, i+j+l )
                        s3 += CHOOSE([k, j]) * POWER([a[1], k-j]) \
                            * POWER([b[1], j]) * s4
                    s2 += CHOOSE([h, i]) * POWER([a[0], h-i]) * POWER([b[0], i]) * s3;
                sl += CHOOSE ([alpha, h]) * CHOOSE ([beta, k]) * CHOOSE ([gamma, m]) \
                    * POWER([vo[0], alpha-h]) * POWER([vo[1], beta-k]) \
                    * POWER([vo[2], gamma-m]) * s2
    c = VECTPROD([a, b])
    if not signed: return sl * VECTNORM(c)
    elif isclose(a[2],0.0) and isclose(b[2],0.0):
        return sl * VECTNORM(c) * SIGN(c[2])
    else: print "error: in signed surface integration"
```

References