Polyhedral geometry 1

Computational Visual Design (CVD-Lab), DIA, “Roma Tre”
University, Rome, Italy

Computational Graphics 2012

Section 1

Linear spaces

Definition

A linear (or vector) space $\mathcal{V}$ over a field $\mathcal{F}$ is a set with two composition rules, such that, for each $\mathbf{u}, \mathbf{v}, \mathbf{w} \in \mathcal{V}$ and for each $\alpha, \beta \in \mathcal{F}$, the rules $+, \cdot$ satisfy the following axioms:

1. $\mathbf{v} + \mathbf{w} = \mathbf{w} + \mathbf{v}$; (commutativity of addition)
2. $\mathbf{u} + (\mathbf{v} + \mathbf{w}) = (\mathbf{u} + \mathbf{v}) + \mathbf{w}$; (associativity of addition)
3. there is a $\mathbf{0} \in \mathcal{V}$ such that $\mathbf{v} + \mathbf{0} = \mathbf{v}$; (neutral el. of addition)
4. there is a $-\mathbf{v} \in \mathcal{V}$ such that $\mathbf{v} + (-\mathbf{v}) = \mathbf{0}$; (inverse of add.)
5. $\alpha \cdot (\mathbf{v} + \mathbf{w}) = \alpha \cdot \mathbf{v} + \alpha \cdot \mathbf{w}$; (distrib. of addition w.r.t. product)
6. $(\alpha + \beta) \cdot \mathbf{v} = \alpha \cdot \mathbf{v} + \alpha \cdot \mathbf{v}$; (distrib. of product w.r.t. addition)
7. $\alpha \cdot (\beta \cdot \mathbf{v}) = (\alpha\beta) \cdot \mathbf{v}$; (associativity of product)
8. $1 \cdot \mathbf{v} = \mathbf{v}$. (neutral element of product)

Example: vector space of real matrices

Let $\mathcal{M}_n^m(\mathbb{R})$ be the set of $m \times n$ matrices with elements in the field $\mathbb{R}$. An element A in such a set is denoted as

$$A = (\alpha_{ij})$$

Addition and multiplication by a scalar are defined component-wise:

$$A + B = (\alpha_{ij}) + (\beta_{ij}) = (\alpha_{ij} + \beta_{ij})$$

$$\gamma A = \gamma(\alpha_{ij}) = (\gamma \alpha_{ij})$$

Example: vector space of polynomials of degree $\leq n$

A linear space we will make often use of in Computer Graphics and Geometric modeling is the space of dimension $n + 1$:

$$\mathcal{P}^n(\mathbb{R}) = \{p : \mathbb{R} \to \mathbb{R} : u \mapsto \sum_{i=1}^{n} a_i p^i, a_i \in \mathbb{R}\}$$

of univariate polynomials of degree $\leq n$ on the real field (with real coefficients), with $p^i \in P_n$, where

$$P_n = (p^n, p^{n-1}, ..., p^1, p^0) \quad \text{and} \quad p^i : u \mapsto u^i$$

is the power basis.

Subspace

Let $(\mathcal{V}, +, \cdot)$ be a vector space on the field $\mathcal{F}$.

$\mathcal{U} \subset \mathcal{V}$ is a subspace of $\mathcal{V}$ if $(\mathcal{U}, +, \cdot)$ is a vector space with respect to the same operations.

$\mathcal{U} \subset \mathcal{V}$ is a subspace of $\mathcal{V}$ if and only if
$\mathcal{U} \neq \emptyset$;
for each $\alpha \in \mathcal{F}$ and $\mathbf{u}_1, \mathbf{u}_2 \in \mathcal{U}$, $\alpha \mathbf{u}_1 + \mathbf{u}_2 \in \mathcal{U}$

codimension of a subspace $\mathcal{U} \subset \mathcal{V}$
is defined as

$$\dim \mathcal{V} - \dim \mathcal{U}$$

Examples of codimension in 1D, 2D, 3D?

Linear combination

Let $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n \in \mathcal{V}$ and $\alpha_1, \alpha_2, \ldots, \alpha_n \in \mathcal{F}$,

The vector

$$\alpha_1 \mathbf{v}_1 + \cdots + \alpha_n \mathbf{v}_n = \sum_{i=1}^{n} \alpha_i \mathbf{v}_i \in \mathcal{V}$$

is called a linear combination of vectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n$ with scalars $\alpha_1, \alpha_2, \ldots, \alpha_n \in \mathcal{F}$

Span

- The set of all linear combinations of elements of a set $S \subset \mathcal{V}$ is a subspace of $\mathcal{V}$.
- Such a subspace is called the span of S and is denoted as

$$\operatorname{lin} S$$

- If a subspace $\mathcal{U}$ of $\mathcal{V}$ can be generated as the span of a set S of vectors in $\mathcal{V}$, then S is called a generating set or a spanning set for $\mathcal{U}$.

Linear independence

- A set of vectors $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$ is linearly independent if

$$\sum_{i=1}^{n} \alpha_i \mathbf{v}_i = \mathbf{0}$$

  implies that $\alpha_i = 0$ for each i
- As a consequence, a set of vectors is linearly independent when none of them belongs to the span of the others.

Bases and coordinates

When working with vector spaces, the concept of basis, a discrete subset of linearly independent elements, is probably the most useful to deal with.

- each element of the space can be represented uniquely as linear combination of basis elements
- this leads to a parametrization of the space, i.e. to represent each element by a sequence of scalars, called its coordinates with respect to the chosen basis.

Bases

A set of vectors $\{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n\}$ is a basis for the vector space $\mathcal{V}$ iff

1. the set is linearly independent, and
2. $\mathcal{V} = \mathrm{lin}\{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n\}$

Bases

- Every two bases of $\mathcal{V}$ have the same number of elements, that is called the dimension of $\mathcal{V}$ and is denoted

$$\dim \mathcal{V}$$

- Some important properties of the bases of a vector space are:
  1. each spanning set for $\mathcal{V}$ contains a basis;
  2. each minimal spanning set is a basis;
  3. each linearly independent set of vectors is contained in a basis;
  4. each maximal set of linearly independent vectors is a basis;

Components

If $(\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n)$ is an ordered basis for $\mathcal{V}$, then for each $\mathbf{v} \in \mathcal{V}$ there exists a unique n-tuple of scalars $\alpha_1, \alpha_2, \ldots, \alpha_n \in \mathcal{F}$ such that

$$\mathbf{v} = \sum_{i=1}^{n} \alpha_i \mathbf{e}_i.$$

Components

The n-tuple of scalars (α_i) is called the components of $\mathbf{v}$ with respect to the ordered basis $(\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_n)$.

- If such a n-tuple were not unique, then $\mathbf{v} = \sum \alpha_i \mathbf{e}_i = \sum \beta_i \mathbf{e}_i$
- But this one would imply $\sum (\alpha_i - \beta_i)\mathbf{e}_i = \mathbf{0}$, hence $(\alpha_i - \beta_i) = 0$,
- i.e. $\alpha_i = \beta_i$, for every i.

Change of basis

- Let $B = (\mathbf{e}_1, \ldots, \mathbf{e}_n) \subset \mathcal{V}$ be a basis for $\mathcal{V}$.
- Of course, their coordinates are $\begin{pmatrix} 1 & 0 & \cdots & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 & \cdots & 0 \end{pmatrix}, \ldots, \begin{pmatrix} 0 & 0 & \cdots & 1 \end{pmatrix}$, and, in B coordinates, the basis is represented by the matrix
$$[B] = [I]$$
.
- If we take n (linearly independent) vectors $V = (\mathbf{v}_1, \ldots, \mathbf{v}_n) \subset \mathcal{V}$, represented in B coordinates as $[V]$, and want to parametrize $\mathcal{V}$ with respect to the new basis, we have, for transformation of coordinates:
$$[I] = [T][V]$$
- and hence:
$$[T] = [V]^{-1}$$

Example: two polynomial bases

- Let $P_3 = (u^3, u^2, u, 1)$
- and $B_3 = ((1-u)^3, 3u(1-u)^2, 3u^2(1-u), u^3)$ be two ordered bases
- for the linear space $\mathcal{P}^3(\mathbb{R})$ of polynomials with deg ≤ 3.
- the $[B_3]$ matrix in the P_3 basis is

$$[B_3]_{P_3} = \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

- the $[P_3]$ matrix in the B_3 basis is

$$[P_3]_{B_3} = [B_3]_{P_3}^{-1} = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1/3 & 1 \\ 0 & 1/3 & 1/6 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

- WHY ?

PLaSM Basics

1. names of functions: all-caps
2. arity: always 1 (number of arguments of functions)
3. dynamic typing
4. higher-level operators (often, non always)
5. small set of predefined operators

PLaSM Basics (AA: Apply-to-All)

```
AA(SUM) [[1,2,3],[4,5,6]]
=> [6,15]
```

PLaSM Basics (DISTL: DISTribute-Left)

```
DISTL [2,[1,2,3]]
// => [[2,1],[2,2],[2,3]]

DISTL [2,[]]
// => []
```

PLaSM Basics (TRANS: TRANSpose)

```
TRANS [[1,2,3],[10,20,30],[100,200,300]]
// => [[1,10,100],[2,20,200],[3,30,300]]

TRANS [[1,2,3,4,5],[10,20,30,40,50]]
// => [[1,10],[2,20],[3,30],[4,40],[5,50]]

TRANS [[],[]]
// => []
```

PLaSM Basics (arithmetic ops)

```
MUL [3,4]
// => MUL [3,4] = 12

MUL [[1,2,3],[4,5,6]]
// => MUL [[1,2,3],[4,5,6]] = [4, 10, 18]
```

```
SUM [3,4]
// => SUM [3,4] = 7

SUM [[1,2,3],[4,5,6]]
// => SUM [[1,2,3],[4,5,6]] = [5, 7, 9]
```

PLaSM Basics (product scalar by vector)

```
PROD [3,[1,2,3]]
// => PROD [3,[1,2,3]] = [3, 6, 9]

PROD [4,[10,20,30]]
// => PROD [4,[10,20,30]] = [40, 80, 120]
```

Plasm.js: Exercise 1 (INNERPROD)

The inner (or scalar) product of $a, b \in \mathbb{R}^m$ is a number

$$\texttt{INNERPROD} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R} : (u, v) \mapsto \sum_{i=1}^{m} \mathbf{u}_i \mathbf{v}_i$$

```
INNERPROD = ([u, v]) -> SUM MUL [u, v]

u = [1,2,3]
v = [10,20,30]
INNERPROD [u, v]
// => 140
```

Plasm.js: Exercise 2 (VECTNORM)

The norm of a vector $a \in \mathbb{R}^m$ is a number.

$$\mathtt{VECTNORM} : \mathbb{R}^m \to \mathbb{R} : v \mapsto \sqrt{\sum_{i=1}^{m} \mathbf{v}_i^2}$$

```
VECTNORM = (v) -> Math.sqrt SUM MUL [v, v]

a = [1,2,3]
VECTNORM a
// => 3.7416573867739413
```

Plasm.js: Exercise 3 (UNITVECT)

The unit vector is a function

$$\mathtt{UNITVECT} : \mathbb{R}^m \to \mathbb{R}^m : v \mapsto \frac{v}{||v||}$$

```
UNITVECT = (v) -> PROD [1/(VECTNORM v), v]

v = [1,2,3]
UNITVECT v
// => [0.2672612, 0.5345224, 0.8017837]

VECTNORM UNITVECT v
// => 1
```

Plasm.js: Exercise 4 (SUM)

SUM adds n vectors in $\mathbb{R}^m$, i.e. the columns of a matrix in $\mathbb{R}^m_n$:

```
a = [1,2,3]
a
// => [1, 2, 3]

b = [10,20,30]
b
// => [10, 20, 30]
```

```
SUM [a,b]
// => SUM [a,b] = [11, 22, 33]
```

Plasm.js: Exercise 5 (SUM)

```
a = [1..10]
a
// => [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

b =  (10*k for k in [1..10])
b
// => [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]

SUM [a,b]
// => [11, 22, 33, 44, 55, 66, 77, 88, 99, 110]
```

```
c =  (100*k for k in [1..10])
SUM [a,b,c]
// => [111, 222, 333, 444, 555, 666, 777, 888, 999, 1110]
```

Plasm.js: Exercise 6 (MATSUM)

Write a function that adds any two matrices $[A], [B]$ (compatible by sum). both $[A], [B]$ must belong to the same linear space $\mathbb{R}^m_n$

```
MATSUM = (args) -> AA(AA(SUM)) AA(TRANS) TRANS args
```

```
A = [ [1,2,3], [4,5,6], [7,8,9] ]
B = [ [10,20,30], [40,50,60], [70,80,90] ]

MATSUM [A,B]
// => [ [11,22,33], [44,55,66], [77,88,99] ]

MATSUM [A,B,A]
// => [ [12,24,36], [48,60,72], [84,96,108] ]

MATSUM [A,B,B,A]
// => [ [22,44,66], [88,110,132], [154,176,198] ]
```

Plasm.js: Exercise 7 (MATPROD)

Write a function that multiplies two matrices (compatible by product)

Remember that

$$A \in \mathbb{R}^m_n, \quad B \in \mathbb{R}^n_p, \quad \text{and} \quad C = AB \in \mathbb{R}^m_p,$$

with

$$C = \left(\ c^i_j\ \right) = \left(\ \mathbf{A^i B_j}\ \right), \qquad 1 \le i \le m, 1 \le j \le p,$$

where $\mathbf{A^i}$ is the i-th row of $\mathbf{A}$, and $\mathbf{B_j}$ is the j-th column of $\mathbf{B}$.