

# 1 Arvutusgeomeetriast

- Geomeetrilised objektid on tundlikud arvutustäpsuse suhtes - ükskõik kui väike arvuline muutus võib mõjutada objektide omavahelisi suhteid põhimõtteliselt (kas sirged lõikuvad, kas punkt asub sirgest "paremal" või "vasakul", jne)
  - eelista arvutamist täpsete (täis)arvudega.
- Ujukomaarvud on ligikaudsed, sama suuruse erineval moel arvutamisel võime saada erineva tulemuse!
  - täpse võrdlemise asemel tuleks väikese erinevuse korral lugeda arvud võrdseks  $abs(x - y) \leq \epsilon$
  - liiga suure või liiga väikese tolerantsi kasutamine võib põhjustada vigu.
- Programmeerija seisukohalt koosnevad geomeetrilised objektid peamiselt eranditest (nullvektor, paralleelsed sirged, kidunud kolmnurk)
  - suurem osa programmist sisaldab erijuhtude käsitlemisest
  - siiski on kasulik teada paari valemit.

# 2 Kasulikud eelteadmised

- punkt tähistame  $(x, y)$
- punktide vaheline kaugus
  - antud kaks punkti:  $(x_1, y_1)$  ja  $(x_2, y_2)$
  - kauguse ruut Pythagorase teoreemi alusel  $d^2 = (x_1 - x_2)^2 + (y_1 - y_2)^2$
  - kauguste omavaheline võrdlemine on sama hea, kui nende ruutude võrdlemine
- vektori mõiste
  - kahe punkti  $(x_1, y_1) \rightarrow (x_2, y_2)$  kaudu defineeritud vektor on  $(x_2 - x_1, y_2 - y_1)$
  - vektori  $V = (v_1, v_2, \dots)$  pikkus  $|V| = \sqrt{v_1^2 + v_2^2 + \dots}$
- determinandi väärtuse leidmine

$$\begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix} = a_1 * b_2 - a_2 * b_1$$

$$\begin{vmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{vmatrix} = a_1 * \begin{vmatrix} b_2 & b_3 \\ c_2 & c_3 \end{vmatrix} - a_2 * \begin{vmatrix} b_1 & c_1 \\ b_3 & c_3 \end{vmatrix} + a_3 * \begin{vmatrix} b_1 & c_1 \\ b_2 & c_2 \end{vmatrix}$$

### 3 Vektorkorrutis

Vektorkorrutise definitsioon kolmemõõtmelise ruumi jaoks, kus  $i, j, k$  on ühikvektorid:

$$u \times v = \begin{vmatrix} i & j & k \\ u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \end{vmatrix}$$

Natuke praktilisem kuju programmeerija jaoks 2-mõõtmelisel tasandil:

$$\begin{vmatrix} 1 & 1 & 1 \\ v_x & v_y & 0 \\ u_x & u_y & 0 \end{vmatrix} = |u| * |v| * \sin(\theta)$$

- vektorkorrutise üheks võimalikuks geomeetriliseks tõlgenduseks on kahe korrutatava vektoriga määratud rööpküliku pindala
- vektorite koordinaatide (ja nendest arvutatud pikkuste) kaudu saame avaldada vektorite vahelise nurga  $\theta$  siinuse.
- kuna  $\sin(\theta)$  on vahemikus  $-\pi/2 \dots \pi/2$  monotoonselt kasvav, siis võime  $\theta$  võrdlemise asendada  $\sin(\theta)$  võrdlemisega!
- $\sin(\theta)$  on vahemikus  $0 \dots \pi$  mittenegatiivne ja  $-\pi \dots 0$  mittepositiivne.
- korrutis nullvektoriga on null
- samasihiliste vektorite korrutis on null

Eelnevat valemit saab teisendada kolme punktiga määratud kolmnurga pindala arvutamiseks mugavamale kujule:

$$S = \frac{1}{2} * \begin{vmatrix} a_x & a_y & 1 \\ b_x & b_y & 1 \\ c_x & c_y & 1 \end{vmatrix}$$

- Vektorkorrutise kaudu leitud pindala võib olla negatiivne - sõltuvalt vektorite valimise järjekorrast.

#### 3.1 Näide - punkti kaugus sirgest

1. valime sirgel kaks (erinevat) punkti, tekib kolmnurk
2. kasutades vektorkorrutist leiame kolmnurga kahekordse pindala  $S$
3. avaldame kolmnurga pindala kõrguse  $h$  (kaugus sirgest) ja aluse  $a$  (valitud kahe punkti vaheline kaugus sirgel) korrutise kaudu
4.  $h = \frac{S}{a}$

### 3.2 Alternatiiv- punkti kaugus sirgest kahendotsinguga

Üldjuhul aeglasem ja ebatäpsem, kuid ei kasuta ruutjuure leidmist

- vali sirgel kaks punkti, uuritavast punktist piisavalt kaugel ja teineteisel pool
- arvuta nende kaugused (ruutjuurt pole vaja) uuritavast punktist
- poolita lõik, arvuta kaugus, võrdle, asenda kaugem otspunkt lõigu keskpunktiga
- kui kallis on üks iteratsiooni samm võrreldes analüütilise lahendusega?
- mitu sammu on vaja teha?

### 3.3 Näide - antud sirge ja kaks punkti.

Kas punktid asuval sama või erineval pool sirget?

1. vali sirgel kaks punkti.
2. arvuta vektorkorrutisega kolnurkade (tippudeks kaks oma valitud ja uuritav punkt) pindalad
3. võrdle pindalade märki.

### 3.4 Näide - suvalise hulknurga pindala arvutamine

hulknurk ei pea olema kumer (miks?), punktid peavad olema järjest.

1. vali suvaline tipp  $p_0$
2.  $S = S_{\Delta}(p_0, p_1, p_2) + S_{\Delta}(p_0, p_2, p_3) + \dots + S_{\Delta}(p_0, p_{n-1}, p_n)$
3. pindala märk sõltub hulknurga tippude järjestamise suunast (päripäeva või vastupäeva)

## 4 Skalaarkorrutis

$$A * B = |A| * |B| * \cos(\theta) = (a_1 * b_1 + a_2 * b_2 + \dots)$$

- kuna  $\cos(\theta)$  on monotoonne vahemikus  $0.. \pi$ , siis sobib skalaarkorrutis vektorite vahelise nurga uurimiseks just selles vahemikus
- samasuunaliste vektorite skalaarkorrutis on positiivne, vastasuunalistel negatiivne - sirgel kolm punkti - milline neist on "keskmine"?
- väärtus on 0, kui vektorid on risti
- $|A| * \cos(\theta)$  on vektori  $A$  projektsioon vektori  $B$  sirgele

## 5 Kahe sirge lõikumine

Antud kaks sirget (kahe punktiga), nende võrrandite üks võimalik üleskirjutamise viis on:

$$(x - x_1) * (y_2 - y_1) = (y - y_1) * (x_2 - x_1)$$

$$(x - x_3) * (y_4 - y_3) = (y - y_3) * (x_4 - x_3)$$

ja meeldejätmiseks sobival kujul lahendivalem:

$$P_x = \frac{\begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} \begin{vmatrix} x_3 & 1 \\ x_4 & 1 \end{vmatrix} - \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix} \begin{vmatrix} x_1 & 1 \\ x_2 & 1 \end{vmatrix}}{\begin{vmatrix} x_1 & 1 \\ x_2 & 1 \end{vmatrix} \begin{vmatrix} y_3 & 1 \\ y_4 & 1 \end{vmatrix} - \begin{vmatrix} y_1 & 1 \\ y_2 & 1 \end{vmatrix} \begin{vmatrix} x_3 & 1 \\ x_4 & 1 \end{vmatrix}}, P_y = \frac{\begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} \begin{vmatrix} y_3 & 1 \\ y_4 & 1 \end{vmatrix} - \begin{vmatrix} y_1 & 1 \\ y_2 & 1 \end{vmatrix} \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix}}{\begin{vmatrix} x_1 & 1 \\ x_2 & 1 \end{vmatrix} \begin{vmatrix} y_3 & 1 \\ y_4 & 1 \end{vmatrix} - \begin{vmatrix} y_1 & 1 \\ y_2 & 1 \end{vmatrix} \begin{vmatrix} x_3 & 1 \\ x_4 & 1 \end{vmatrix}}$$

- valemis võib tekkida nulliga jagamine.
- paralleelsed sirged?
- kattuvad sirged?

### 5.1 Näide: antud kaks lõiku.

Leida nende ühine punkt (kui selline on).

1. kasutame valemit
2. töötleme erijuhud.
  - kaks punkti
  - punkt ja lõik
  - sirgete lõikumine ei tähenda lõikude lõikumist
  - lõigud samal sirgel (2 erijuhtu)

## 6 Algoritm: Punkti hulga kumer kate.

- leia minimaalse  $y$  (ja nende hulgas minimaane  $x$ ) koordinaadiga punkt  $P_0$ , lisa punkt  $P_0$  vastusesse.

- sorteeri punktid  $P_1 \dots P_n$   $X$  telje ja  $P_0 -> P_i$  vektori suuna vahelise nurga alusel (skalaarkorrutis!)
- lisa vastusesse punkt  $P_1$
- lisa järjest vastusesse punktid  $P_2 \dots P_n$ , veendudes igal sammul, et moodustuv hulknurk on kumer (st pööre toimub vasakule) (vektorkorrutis!) vajadusel eemalda eelmisi punkte nii kaua, et saaksid reeglit rikkumata jooksva punkti lisada viimaseks.
- lõpeta, kui oled edukalt lisanud punkti  $P_n$
- tähelepanu punktidega mis asuvad hulnurga servadel, kuid ei ole tippudeks.