Robusthed i geometriske algoritmer

Michael Neidhardt

18. december 2008

Teori:
Reel RAM $\Rightarrow$ reelle tal og uendelig præcision.
Data i generel position.
O(1) tid pr. basal regneoperation.

Praksis:
Endelig præcision.
Flydende tal $\Rightarrow$ afrundingsfejl.
Data ikke i generel position.

Eksperimenter med algoritmer der finder alle nærmeste naboer for en punktmængde i planen:

Delaunay-triangulering: TTL & CGAL.
Plane sweep.
Kd-træ.

Både hele og flydende tal anvendes.

To centrale funktioner, kaldet prædikater:

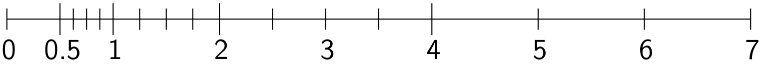
orient2d: Udgør tre punkter højre/venstresving eller er de på linie.
incircle: Er et punkt uden for, inde i eller på cirklen gennem tre andre punkter.

Begge baseret på fortegn for determinant.

Hertil kommer funktioner der danner nye geometriske objekter, fx skæringspunkter.

Flydende tal som defineret af IEEE 754 er en endelig mængde, $\mathbb{F}$.
Et $\tilde{x} \in \mathbb{F}$ består af: $\pm s \times 2^e$.

Minisystem med præcision $= 3, e_{min} = -1, e_{max} = 2$:



Maskinpræcisionen ε angiver den største fejl. I dobbelt præcision og med afrunding til nærmeste er $\varepsilon = 2^{-53}$.

For et reelt tal x og et flydende tal $\tilde{x}$ gælder:

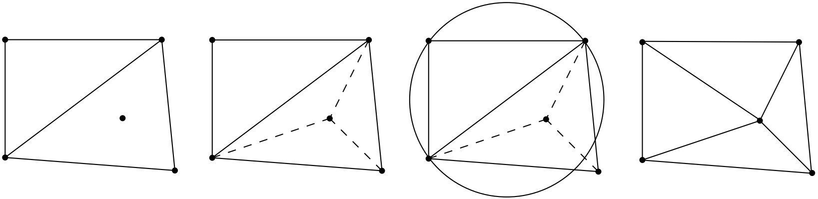
$$|x - \tilde{x}| \leq \varepsilon|x| \quad \Leftrightarrow \quad \left|\frac{x-\tilde{x}}{x}\right| \leq \varepsilon.$$

Fejltolerante metoder — bla. ε-*tweaking*, 'tykke' objekter, topologi-orienteret.

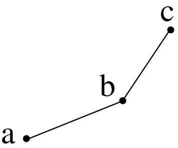
Eksakte metoder — filtre, fejlanalyse, multipræcisions-typer, intervalregning.

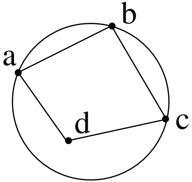
Perturbering af data $\Rightarrow$ generel position.

Inkrementel metode: Omslut punktmængde med fiktiv trekant, punktlokalisering, kantlovliggørelse, fjernelse af fiktiv trekant.



Punktlokalisering løses med orient2d.
Kantlovliggørelse løses med incircle.





Orient2d:

$$\mathbf{D} = \begin{vmatrix} a_x & a_y & 1 \\ b_x & b_y & 1 \\ c_x & c_y & 1 \end{vmatrix} = \begin{vmatrix} a_x - c_x & a_y - c_y \\ b_x - c_x & b_y - c_y \end{vmatrix}$$

$\mathbf{D} = 0 \Rightarrow$ på linie, ellers venstre- eller højresving.
Hvis $|\mathbf{D}| >$ maksimale fejl $\Rightarrow$ fortegn($\mathbf{D}$) OK,
Ellers benyt dyrere metode.

Filtre til eksakt beregning af fortegn for determinanter:

Shewchuk: Fire trin med stigende præcision: Fra flydende tal til eksakt beregning.

CGAL: Test for underløb, overløb og determinantens størrelse. Ved fejl benyttes evt. LU-faktorisering med intervalregning og som sidste udvej multipræcisionstype.

ABDPY: Rækkeoperationer formindsker værdierne i matricen (2x2).

Shewchuk: Ekspansioner sikrer multipræcision. En sum/differens kan beregnes som to led, det flydende og fejlen.

Eksempel: Kvadreret afstand mellem to punkter, a og b:
$kdist(a, b) = (a_x - b_x)^2 + (a_y - b_y)^2$.

Ekspansionen $(x, y) = \text{TODIFF}(a_x, b_x)$ hvor $y < \varepsilon(a_x - b_x)$.

$kdist(a, b) = (x_1 + y_1)^2 + (x_2 + y_2)^2 =$

$$x_1^2 + y_1^2 + 2x_1y_1 + x_2^2 + y_2^2 + 2x_2y_2 =$$

$$\underbrace{x_1^2 + x_2^2}_{A} + \underbrace{2x_1y_1 + 2x_2y_2}_{B} + \underbrace{y_1^2 + y_2^2}_{C}$$

CGAL Filtreret kerne. 1. trin i filtret (orient2d):

$$\mathbf{D} = \begin{vmatrix} a_x - c_x & a_y - c_y \\ b_x - c_x & b_y - c_y \end{vmatrix} = ac_x bc_y - ac_y bc_x.$$

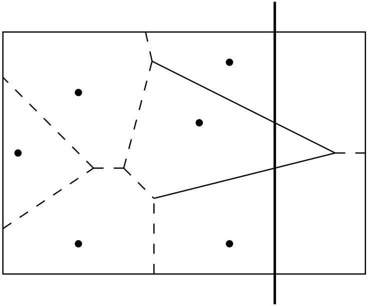
maxfejl $= 2^{-50} \times$max-værdi for $\mathbf{D}$.

1: Risiko for underløb og $\neq 0 \Rightarrow$ dyrere metode.

2: Overløb $\Rightarrow$ dyrere metode.

3: $|\mathbf{D}| \leq$ maxfejl $\Rightarrow$ dyrere metode.

Man fejer over punktmængden og vedligeholder et delvist Voronoi-diagram.



Prioritetskø indeholder punkter (hændelserne).
Binært søgetræ indeholder aktive bisektorer, sorteret på skæring med fejelinien.

Algoritmen kræver heltalsinddata.

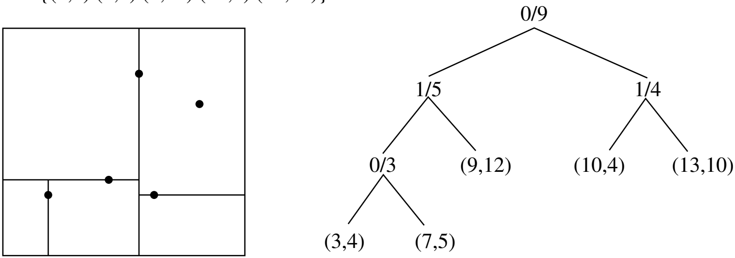
Numerisk robusthed: Maksimale præcision udregnes. I praksis ofte lig med multipræcision.

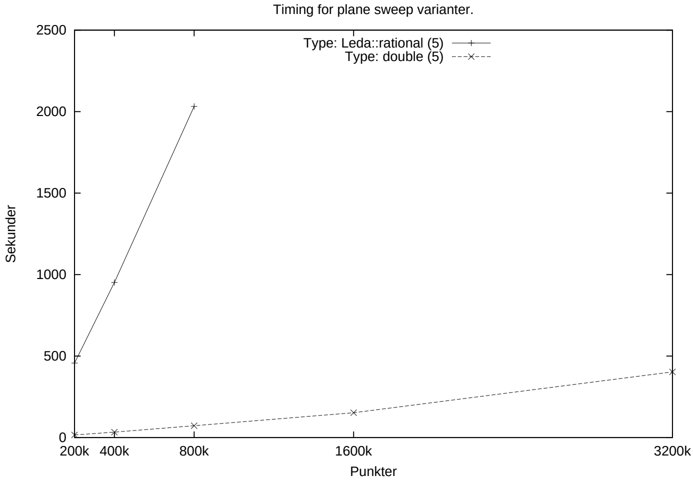
Problemer ved dannelse af skæringspunkter — enten mellem bisektorer eller mellem bisektor og fejelinie. Løses ved at benytte homogene koordinater eller ved *snap rounding*.

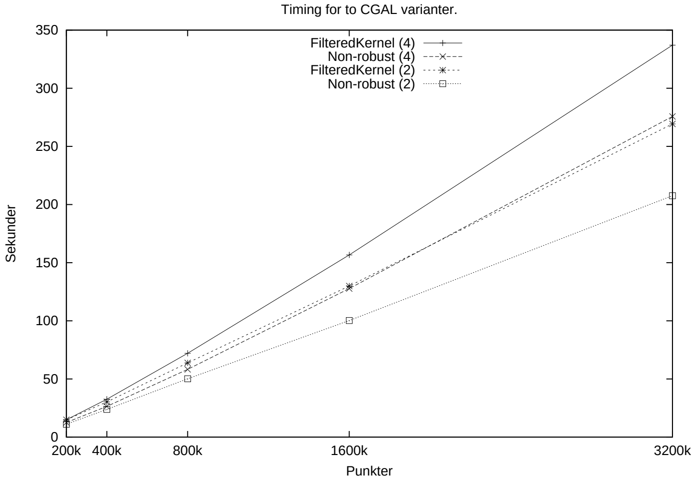
Større praktisk problem er måske algoritmisk robusthed.

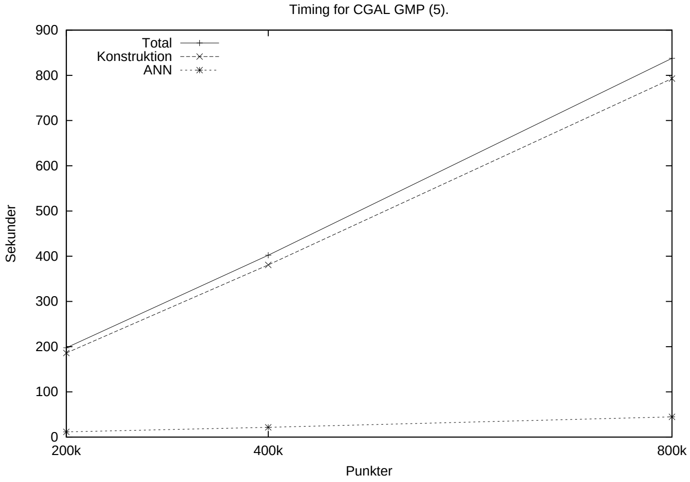
Simpel og effektiv til løsning af alle-nærmeste-naboer.
Ingen robusthedsproblemer.
Overløb kan dog forekomme.
Tilsyneladende svært at garantere maksimal søgetid.

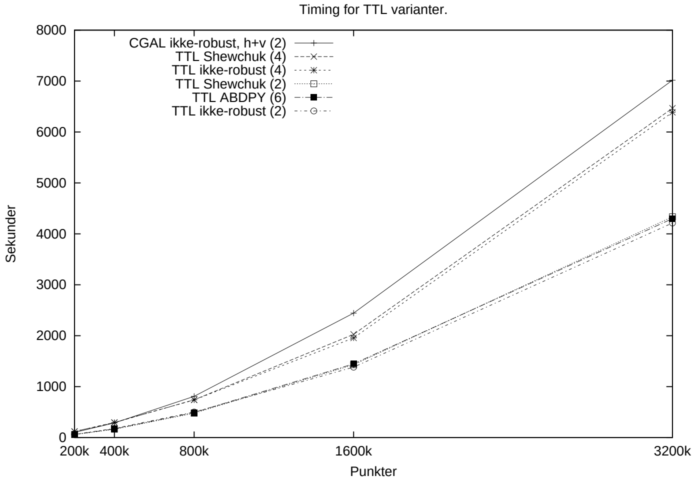
S = {(3,4) (7,5) (9,12) (10,4) (13,10)}

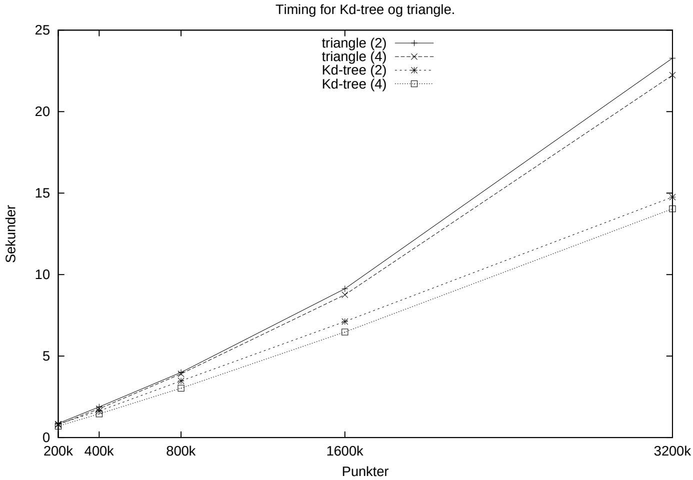



Timing for plane sweep varianter.
Type: Leda::rational (5)
Type: double (5)
Sekunder
2500
2000
1500
1000
500
0
200k
400k
800k
1600k
3200k
Punkter


Timing for to CGAL varianter.
FilteredKernel (4)
Non-robust (4)
FilteredKernel (2)
Non-robust (2)
Sekunder
350
300
250
200
150
100
50
0
200k
400k
800k
1600k
3200k
Punkter


Timing for CGAL GMP (5).
Total
Konstruktion
ANN
900
800
700
600
500
400
300
200
100
0
Sekunder
200k
400k
800k
Punkter


Timing for TTL varianter.
CGAL ikke-robust, h+v (2)
TTL Shewchuk (4)
TTL ikke-robust (4)
TTL Shewchuk (2)
TTL ABDPY (6)
TTL ikke-robust (2)
Sekunder
0
1000
2000
3000
4000
5000
6000
7000
8000
200k
400k
800k
1600k
3200k
Punkter


Timing for Kd-tree og triangle.
triangle (2)
triangle (4)
Kd-tree (2)
Kd-tree (4)
Sekunder
0
5
10
15
20
25
200k
400k
800k
1600k
3200k
Punkter

Program	Tilfældige punkter i enhedskvadratet	KMPSY	KMPSY-2
TTL Ikke-robust	4.11	24.10	26.96
TTL Filter (Shewchuk)	3.80	21.98	22.69
CGAL Ikke-robust	1.81	Fejler	13.86
CGAL Filter	2.29	39.64	16.03
Kd-træ	0.12	0.25	0.22
triangle	1.10	1.43	1.34

Tabel: Tidsforbrug i sekunder ved ANN-beregning for tre mindre punktmængder (ca. 32898 punkter).

Prædikat	Tilfældige punkter i enhedskvadratet	KMPSY	KMPSY-2
orient2d A	2961298	98389	72713
orient2d B	0	574	1225
orient2d C	0	510	1223
orient2d D	0	2	65
incircle A	510856	5622041	5617723
incircle B	0	33115	33989
incircle C	0	759	521
incircle D	0	255	0

Tabel: Antal kald til orient2d og incircle i TTL Filter (Shewchuk).

Prædikat	Tilfældige punkter i enhedskvadratet	KMPSY	KMPSY-2
orient2d A	1836436	8134952	10795812
orient2d B	0	485306	1358
incircle A	304512	5605748	5540437
incircle B	0	97706	32426

Tabel: Antal kald til orient2d og incircle i CGAL Filter.

Filtre foretrækkes hvis tilgængelige.
Uklart hvilket af de anvendte filtre er bedst. Også situationsafhængigt.

Typiske fælder er determinanter, kvadratrødder, konstruktioner som fx skæringspunkter.

Ikke trivielt at fremprovokere disse fejl.