ANALISIS AKURASI PERPOTONGAN GARIS DENGAN LINGKARAN DAN LINGKARAN DENGAN LINGKARAN

Djoni Haryadi Setiabudi
Fakultas Teknik, Jurusan Teknik Informatika – Universitas Kristen Petra
e-mail: djonihs@peter.petra.ac.id

ABSTRAK : Untuk meneliti masalah akurasi perpotongan antara garis dan lingkaran dan antara dua lingkaran dapat digunakan metode unified representation. Ada dua formulasi yang digunakan untuk perpotongan antara garis dan lingkaran. Formula pertama menggunakan titik awal pada garis yang berada diluar lingkaran, formulasi yang kedua menggunakan titik tengah antara kedua titik potong sebagai titik awalnya. Untuk perpotongan antara dua lingkaran, ada tiga formulasi. Formulasi yang pertama menggunakan sumbu radikal yang memotong salah satu lingkaran. Formulasi yang kedua menggunakan sumbu radikal yang memotong lingkaran ortogonal yang diperoleh dengan penambahan faktor beban pada persamaan dua lingkaran. Formulasi yang ketiga menggunakan sumbu radikal yang memotong lingkaran ortogonal yang persamaannya didapatkan dengan menambahkan persamaan dari dua lingkaran.

Kata kunci : komputer grafik, algoritma, garis, lingkaran, akurasi.

ABSTRACT : *The problem in accuracy of intersection between a circle and a straight line and between two circles was considered using a unified representation. There are two formulations of the intersection between a straight line and a circle. The first formulation uses an initial point of the line outside the circle, the second formulation uses the midpoint of the line between the intersection points as the initial point. For circle-circle intersection, there are three formulations. The first formulation uses the radical axis intersecting with one of the two circles. The second formulation uses the radical axis intersecting with the orthogonal circle obtained by adding the weighting equations of the two circles. The third formulation uses the radical axis intersecting with the orthogonal circle obtained by adding the equations of the two circles.*

Keywords : computer graphics, algorithms, line, circle, accuracy

1. PENDAHULUAN

Pada software-software untuk aplikasi grafik dan CAD ketelitian merupakan salah satu faktor yang sangat penting. Pada penelitian ini pembahasan difokuskan pada perpotongan antara garis dan lingkaran, karena kedua grafik primitive ini yang sering dijumpai dalam aplikasi grafik. Karena toleransi yang diberikan harus cukup besar untuk meyakinkan agar semua kemungkinan kasus khusus dapat dideteksi, beberapa kasus dapat salah diinterpretasikan , yang dapat mengurangi ketelitian. Untuk ini perlu dibuat suatu cara untuk mengatasi hal ini, khususnya pada algoritma sederhana yang umumnya mengalami kegagalan. Kegagalan ini biasanya disertai dengan penurunan akurasi pada saat kasus-kasus khusus dijumpai.

2. ANALISIS [3]

Representasi suatu lingkaran dengan menggunakan pusat dan jari-jari dapat berubah menjadi suatu titik tunggal (jari-jari 0), tetapi tidak dapat berubah menjadi garis lurus (jari-jari tak berhingga). Untuk mengatasi hal ini perlu dicari suatu represantasi yang memungkinkan terjadinya kedua perubahan diatas khususnya untuk kasus terjadinya perpotongan.

Suatu lingkaran adalah sekumpulan titik-titik **p** pada bidang Euclidean yang memenuhi persamaan dalam bentuk :

$$a\,\mathrm{p.p} + \mathrm{b.p} + \mathrm{c} = 0 \qquad (1)$$

Suatu lingkaran dapat direpresentasikan oleh skalar a dan c, dan sebuah vector **b**, yang memenuhi persamaan $\mathbf{b.b} - 4ac \geq 0$. Harga positif dari diskriminan adalah k=+_(**b.b** - 4ac). Maka, suatu lingkaran dapat direpresentasikan dengan 5 tuple {a,**b**,c,k}. Titik pusat **q** dan jari-jari r dari lingkaran {a,b,c,k} adalah :

$$q = \frac{-b}{2a} \quad r = \frac{k}{2a} \qquad (2)$$

Apabila jari-jarinya nol (k=0), lingkaran berubah menjadi titik tunggal -**b**/2a dan bila jari-jarinya tak berhingga (a=0), persamaan 1 dapat difaktorisasi menjadi dua suku linier:

$$(p.v + 1)(p.b + c) = 0 \qquad (3)$$

2.1 Perpotongan antara garis dengan lingkaran

Titik *a* pada garis **n.p** + *d* = 0 memenuhi persamaan parametrik :

$$p(t) = -t\,Rot(n) - p(0) \qquad (4)$$

dimana : Rot(**n**) = $(-n_y, n_x)$, yaitu vektor **n** yang diputar 90 derajat berlawanan jarum jam dan:

$$p(0) = -\frac{d\,n}{n\,.\,n} \qquad (5)$$

Dengan mensubstitusi persamaan 5 ke persamaan 1, titik-titik pada garis dan lingkaran memenuhi persamaan kuadrat dalam t :

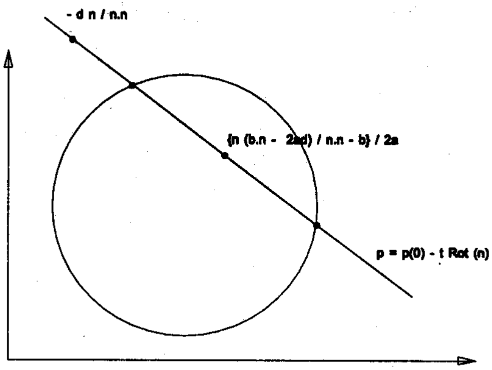
$$A\,t^2 + B\,t + C = 0 \qquad (6)$$

dimana :
A = a (n.n)(Rot(n).Rot(n)),
B = -b.Rot(n)(n.n),
C = a d^2 - (b.n)d + c (n.n)

Persamaan ini tetap berlaku meskipun lingkaran berubah nejadi titik maupun garis. Pada saat berubah menjadi garis, persamaan berubah menjadi persamaan linier **b**.Rot(**n**)(**n.n**)t = **b.n**d-c(**n.n**). Pada saat berubah menjadi titik, persamaan tetap berupa persamaan kuadrat dengan diskriminan, yang selalu lebih keci atau sama dengan nol. Untuk mendapatkan solusi dari titik-titik potong, nilai penyelesaian untuk t disubstitusikan ke persamaan 4.

Error relatif pada parameter *t* menimbulkan error pada titik potong yang akan bertambah besar pada saat semakin jauh dari titik pusat. Ini adalah hasil yang didapat dari pemilihan titik **p**(0) = -**dn/n.n** sebagai titik yang terdekat ke titik pusat. Alternatif lain didapat dengan memilih titik tengah garis antara titik-titik potong (lihat gambar 1) :

$$p(0) = \frac{\{\, n(b.n - 2ad)/n.n - b \,\}}{2a} \qquad (7)$$



Gambar 1. Titik tengah garis diantara dua titik potong

Tangency harus dapat dideteksi pada saat lingkaran berubah menjadi garis. Untuk meyakinkan bahwa *tangency* dapat dideteksi, penyelesaian dari persamaan 7 yang mempunyai bagian imajiner harus dianggap sebagai penyelesaian yang berimpit. Jarak ekstrim dari sebuah titik ke lingkaran dapat dihitung sebagai berikut :

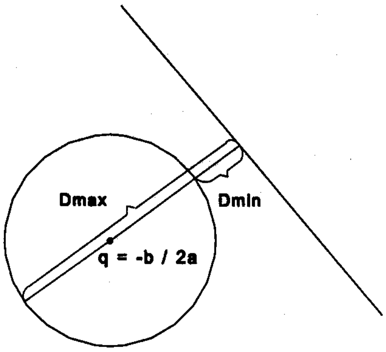
a). Dicari jarak antara titik pusat lingkaran dengan garis :

$$d_i = \frac{n.p + d}{\sqrt{n.n}} \qquad (8)$$

b) Dihitung jarak maksimum dan minimum dengan :

$$D_{min} = d_i - r$$
$$D_{max} = d_i + r \qquad (9)$$

dimana r adalah jari-jari lingkaran dari persamaan 3. Lihat gambar 2.



Gambar 2. Jarak minimum dan maksimum antara pusat lingkaran dan garis lurus

Akhirnya, dengan mensubstitusi titik pusat dan jari-jari lingkaran, jarak ekstrim dapat dituliskan sebagai berikut :

$$D_{\min} = \left| \left(\left| \frac{d - \frac{b.n}{2a}}{\sqrt{n.n}} \right| - \left| \frac{k}{2a} \right| \right) \right| \qquad (10)$$

dan

$$D_{\max} = \left| \left(\left| \frac{d \; \frac{b.n}{2a}}{\sqrt{n.n}} \right| + \left| \frac{k}{2a} \right| \right) \right| \qquad (11)$$

Jika kondisi *tangency* tidak dipenuhi, persamaan 7 dapat diselesaikan dengan metode yang biasa untuk memberikan penyelesaian atau nilai parameter :

$$t_1 = \frac{(-B - \sqrt{K})}{2A} \qquad t_2 = \frac{2C}{(-B - \sqrt{K})} \qquad (12)$$

Jika B≥0, ekspresi tersebut menghindari perbedaan antara dua nilai yang sama, sehingga memberikan hasil yang lebih akurat dari formula konvensional untuk penyelesaian persamaan kuadrat; jika B<0, didapat persamaan yang sebanding :

$$t_1 = \frac{2C}{(B + \sqrt{K})} \qquad t_2 = \frac{(-B + \sqrt{K})}{2A} \qquad (13)$$

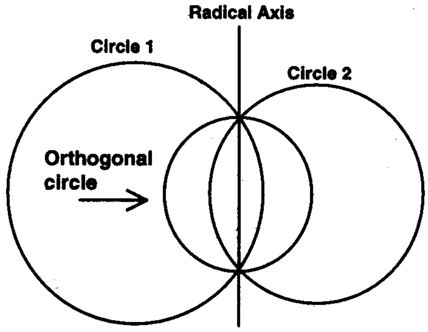
2.2 Perpotongan dua lingkaran

Perhatikan dua lingkaran $\{a_i, b_i, c_i, k_i\}$, i= 1,2. Sumbu radikal (garis yang ditarik melalui dua titik potong) diperoleh dengan mengurangkan persamaan 1 :

$$(a_2 b_1 - a_1 b_2).p + (a_2 c_1 - a_1 c_2) = 0 \qquad (14)$$

Titik-titik potong lingkaran dengan lingkaran dapat diperoleh dari perpotongan sumbu radikal dengan salah satu dari dua lingkaran (lihat gambar 3). Cara lain untuk mendapatkan titik potong adalah dengan memotongkan sumbu radikal dengan lingkaran ortogonal yang diperoleh dengan menambahkan persamaan dari dua lingkaran sebagai berikut :

$$(a_1 + a_2) p.p + (b_1 + b_2).p + c_1 + c_2 = 0 \qquad (15)$$



Gambar 3. Perpotongan antara sumbu radikal dengan lingkaran ortogonal

Alternatif lain untuk menyatakan lingkaran ortogonal adalah dengan menambahkan faktor beban:

$$2 a_1 a_2 p.p + (a_2 b_1 + a_1 b_2).p + (a_2 c_1 + a_1 c_2) = 0 \qquad (16)$$

3. IMPLEMENTASI [5]

3.1 Akurasi mesin

Akurasi mesin T dapat dicari dengan menggunakan testing 1 + T = 1. Algoritmanya dapat dilihat pada Algoritma 1.

Algoritma 1 :

```
Epsilon := 1;
while (1 + Epsilon) <> 1 do
   Epsilon := Epsilon / 2;
```

3.2 Implementasi Perpotongan Garis dan Lingkaran

Implementasi ini menggunakan persamaan 6 untuk garis dan persamaan 1 untuk lingkaran. Untuk persamaan garis, dipilih $n_x = 2$ dan $n_y = 1$ dan untuk persamaan garis, jari-jari r = 50 dan pusatnya **q**=(10,10).

Persamaan 8 digunakan untuk menentukan titik perpotongan. Untuk implementasi pencarian titik potong p(0) diperoleh dari persamaan 7 dan untuk koefisien A, B and C, digunakan persamaan 9. Untuk menghitung nilai parameter t_1 dan t_2, digunakan persamaan 15, dimana B>=0. Algoritma perpotongan garis-lingkaran ini diformulasikan pada Algoritma 2.

Algoritma 2 :

```
procedure Intersection_Line_Circle;
 begin
  t1 := 2 * C / ( -B - %K );
  p0(x) := -d * n(x) / n.n;
  p0(y) := -d * n(y) / n.n;
  x1 := -t1 * Rot_n(x) + p0(x);
  y1 := -t1 * Rot_n(y) + p0(y);
  t2 := ( -B - %K ) / ( 2 * A );
  x2 := -t2 * Rot_n(x) + p0(x);
  y2 := -t2 * Rot_n(y) + p0(y);
 end;
```

3.3 Implementasi Perpotongan Lingkaran dan Lingkaran

Impelementasi yang dilakukan menggunakan jari-jari r1 = 60, pusat **q1** =(10,10) untuk lingkaran 1 dan jari-jari r2 = 50, pusat **q2** = (-30,-30) untuk lingkaran 2.

Untuk mencari titik potong lingkaran dengan lingkaran, digunakan radical axis yaitu garis yang menghubungkan titik-titk potong. Kemudian titik-titik potong dapat ditemukan dengan mancari titik potong dengan salah satu dari lingkaran 1 atau lingkaran 2. Jika persamaan untuk lingkaran 1 adalah a1 **p.p** + **b1.p** + c1 = 0 dan persamaan untuk lingkaran 2 adalah a2 **p.p** + **b2.p** + c2 = 0, persamaan untuk radical axis adalah persamaan 17. Algoritma untuk menghitung titik-titik potong ini idapat dilihat pada Algoritma 3.

Algoritma 3 :

```
procedure Intersection_Circle_Circle;
begin
 t1 := 2 * C / (-B - %K);
 p0(x) := -d * n(x) / n.n;
 p0(y) := -d * n(y) / n.n;
 p(x) := -t1 * Rot_n(x) + p0(x);
 p(y) := -t1 * Rot_n(y) + p0(y);
 t2 := (-B - %K) / (2 * A);
 p(x):= -t2 * Rot_n(x) + p0(x);
 p(y):= -t2 * Rot_n(y) + p0(y);
end;
```

4. HASIL PENGUJIAN

Tujuan pengujian ini adalah untuk meneliti ketelitian hasil perpotongan seperti dijelaskan pada bagian sebelumnya. Hasilnya tergantung dari ketelitian mesin (*machine accuracy*) dan bahasa pemrograman yang digunakan, dalam hal ini adalah Turbo Pascal.

4.1 Experimen untuk Perpotongan Garis dan Lingkaran

Experimen ini menggunakan persamaan 5 untuk garis dan persamaan 1 untuk lingkaran. Untuk persamaan garis, dipilih $n_x = 2$ dan $n_y = 1$ dan untuk persamaan lingkaran, jari-jari r=50 dan pusat **q** =(10,10). Untuk mendapatkan titik potong digunakan persamaan 7. Ada dua macam experimen untuk mendapatkan titik potong. Pertama, p(0) didapatkan dari persamaan 6 dan untuk koefisien A, B dan C digunakan persamaan 8. Untuk menghitung nilai parameter t_1 dan t_2 digunakan persamaan 14 dimana B>=0.

Hasil experimen dapat dilihat pada Tabel 1 untuk titik potong yang pertama dan Tabel 2 untuk titik potong yang kedua. Jarak minimum menunjukkan jarak minimum antara pusat lingkaran dan garis.

4.2 Experimen untuk Perpotongan Lingkaran dengan Lingkaran

Experimen ini menggunakan jari-jari r1 = 60, pusat **q1** =(10,10) untuk lingkaran 1 dan jari-jari r2 = 50, pusat **q2** =(-30,-30) untuk lingkaran 2.

Ada tiga cara untuk menentukan titik potong. Pertama, digunakan sumbu radikal yaitu garis yang menghubungkan titik-titik potong. Kemudian titik potong dapat ditemukan dengan memotongkan sumbu radikal dengan salah satu diantara lingkaran 1 atau lingkaran 2. Jika persamaan untuk lingkaran 1 adalah a1 **p.p** + **b1.p** + c1 = 0 dan persamaan untuk lingkaran 2 adalah a2 **p.p** + **b2.p** + c2 = 0, maka persamaan untuk sumbu radikal adalah persamaan 16. Hasil dari experimen ini dapat dilihat pada Tabel 3 dan Tabel 4.

Hasil perpotongan berbeda kecuali untuk extended datatype. Dari hasil ini dapat disimpulkan bahwa titik potong menggunakan extended datatype merupakan hasil yang valid, sehingga persentasi error pada datatype yang lain dapat dihitung.

Untuk mendapatkan tangency, digunakan tes Dmin < Tol, dimana Dmin menyatakan jarak antara sumbu radikal dan lingkaran. Sama pada experimen untuk garis-lingkaran untuk tangency, lingkaran 1 digerakkan ke

Tabel 1.Titik Potong Pertama dari Perpotongan Garis dan Lingkaran dengan menggunakan Formulasi p(0) pertama

DATATYPE	t	JARAK MIN	X	Y
REAL	18.88061301782	32.111456179991364	6.880613017827272	-23.761226035654544
SINGLE	18.88061332702	32.111457824707031	6.880615234375000	-23.761226654052734
DOUBLE	18.88061301782	32.111456180001681	6.880613017821104	-23.761226035642202
EXTENDED	18.88061301782	32.111456180001682	6.880613017821100	-23.761226035642200

Tabel 2.Titik Potong Kedua dari Perpotongan Garis dan Lingkaran dengan menggunakan Formulasi p(0) pertama

DATATYPE	t	JARAK MIN	X	Y
REAL	-22.88061301782	32.111456179991364	5.1193869821727275	59.7612260356545448
SINGLE	-22.88061332702	32.111457824707031	5.1193866729736328	59.7612266540527344
DOUBLE	-22.88061301782	32.111456180001681	5.1193869821788986	59.7612260356422027
EXTENDED	-22.88061301782	32.111456180001682	5.1193869821788996	59.7612260356422007

Tabel 3. Perpotongan Lingkaran-lingkaran : Sumbu Radikal - Lingkaran 1 Titik Potong Pertama

DATATYPE	T	x	y
REAL	4924.32974499464035	-49.7038649666355923	15.9538649666937999
SINGLE	4924.32910156250000	-49.7038612365722656	15.9538660049438477
DOUBLE	4924.32974500083674	-49.7038649666722421	15.9538649666722421
EXTENDED	4924.32974500083615	-49.7038649666722410	15.9538649666722410

Tabel 4. Perpotongan Lingkaran-lingkaran : Sumbu Radikal - Lingkaran 1 Titik Potong Kedua

DATATYPE	T	x	y
REAL	-4924.32974499464035	15.9538649666937999	-49.7038649666355923
SINGLE	-4924.32910156250000	15.9538660049438477	-49.7038612365722656
DOUBLE	-4924.32974500083674	15.9538649666722421	-49.7038649666722421
EXTENDED	-4924.32974500083615	15.9538649666722410	-49.7038649666722410

lingkaran 2 dengan penambahan yang sama. Titik pusat lingkaran 1 sebelum digerakkan adalah (47.8,47.8).

5. KESIMPULAN

5.1 Perpotongan garis dengan lingkaran

Ada dua cara untuk mendapatkan titik potong antara garis dengan linngkaran. Hasil implementasi menggunakan berbagai lingkaran dengan jari-jari dan titik pusat yang berbeda, memberikan hasil yang sama, kecuali untuk parameter t. Untuk implementasi dengan tangency, dua pendekatan tersebut juga memberikan titik tangency yang sama untuk semua datatype yang digunakan. Karena pendekatan pertama memerlukan perhitungan numerik yang lebih sedikit, pendekatan ini lebih disarankan untuk penggunaan yang praktis.

5.2 Perpotongan lingkaran dengan lingkaran

Ada tiga alternatif untuk mendapatkan perpotongan antara lingkaran dengan lingkaran. Dari hasil implementasi, dapat dilihat bahwa hasil dari pendekatan pertama dan ketiga sama, sedangkan yang kedua sedikit berbeda. Hal ini karena toleransi dari pendekatan kedua lebih besar dari yang pertama dan ketiga. Dapat disimpulkan, bila ketelitian tinggi diperlukan, pendekatan pertama atau ketiga perlu dipilih. Tetapi pendekatan kedua lebih efisien dari yang lain, karena tidak memerlukan perhitungan Dmin untuk mengecek terjadinya tangency. Untuk penggunaan yang praktis, pendekatan yang kedua lebih disarankan.

DAFTAR PUSTAKA

1. Chasen, S.H., *Geometric Principles and Procedures for Computer Graphic Applications*, Prentice Hall, Engelwood Cliffs, New York, 1978.

2. Foley, J.D. and Van Dam, A., *Fundamentals of Interactive Computer Graphics*, Addison Wesley, 1982.

3. Middleditch, A.E., Stacey, T.W. and TOR, S.B., Intersection Algorithms for Lines and Circles, *ACM Transactions on Graphics,* Vol. 8, No. 1, 1989, pp. 25-40.

4. Rogers, D.F. and Adams, J.A., *Mathematical Elements for Computer Graphics,* Second Edition, Mc. Graw-Hill, Inc. 1990.

5. Setiabudi, Djoni H., *Generation of Circular and Elliptical Arcs*, Masters thesis, Asian Institute of Technology, Thailand, 1991.