Penggunaan Metode Numerik untuk Integrasi dalam Mencari Nilai *e*

Christianto - NIM : 13510003[1]
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
[1]13510003@std.stei.itb.ac.id, handojo_christianto@yahoo.com

***Abstract*—Makalah ini akan membahas penggunaan beberapa metode integrasi dalam mencari nilai e (bilangan natural) yang tingkat akurasinya dapat disesuaikan sesuai dengan keperluan. Kemudian makalah ini juga menelusuri beberapa kemungkinan metode perhitungan lain berdasarkan pada definisi nilai e dan membandingkan efektivitas metode integrasi dengan metode lainnya dalam menghitung nilai e.**

***Index Terms*—e, metode integrasi,bilangan natural.**

1. Pendahuluan

Konstanta *e*, juga disebut sebagai bilangan natural, merupakan salah satu konstanta yang paling penting dalam dunia matematika. Sifat-sifat unik yang dimilikinya membuatnya sering muncul dalam persamaan matematika, terutama solusi-solusi dari integrasi persamaan dan juga dalam pemanfaatannya sebagai basis dari logaritma natural. Sayangnya, salah satu sifat *e* membuat pencarian nilai eksak dari e menjadi mustahil, karena tidak ada pola pengulangan digit yang muncul. Satu-satunya yang dapat dilakukan dalam mencari nilai pasti dari *e* adalah menghitungnya hingga digit tertentu di belakang koma. Penjelasan mengenai *e* yang cukup detail dapat ditemukan pada [2].

Makalah ini akan mencoba memakai beberapa teknik integrasi yang sering dipakai, yaitu teknik integrasi trapesium, metode Simpson 1/3, serta metode Romberg untuk mencari pendekatan nilai *e*. Kemudian makalah ini juga akan menggunakan definisi lain dari nilai *e* untuk mencari pendekatan nilai *e* dan membandingkan hasilnya dengan hasil dari metode-metode integrasi yang dipakai.

2. Mencari Nilai *e* dengan Metode Integrasi

Mencari nilai *e* dengan metode integrasi didasarkan pada salah satu definisi nilai *e* berdasarkan teori kalkulus, yaitu :

$$\int_{1}^{e} 1/x\, dx = 1$$

Persamaan diatas dapat dibuktikan dengan menhitungnya secara langsung, karena integral dari 1/x dx adalah ln(x), dimana ln berarti logaritma natural dari x. Karena ln(e) adalah 1 dan ln(1) adalah 0, maka hasil integrasinya adalah 1.

Dalam perhitungan yang dipakai berikutnya ini, hasil integrasi dihitung menggunakan metode numerik untuk integrasi, karena definisi nilai ln sendiri terkait erat dengan nilai *e* yang dicari. Metode-metode integrasi yang dipakai semua berpaku dengan cara menghitung nilai fungsi pada titik-titik tertentu dan karenanya tidak terpengaruh oleh nilai *e* yang sebenarnya.

Selain memakai metode-metode integrasi untuk menghitung luas daerah, metode bagi dua juga digunakan untuk mencari nilai *e* yang tepat hingga ketelitian tertentu.

3. Metode Integrasi

Dalam makalah ini, akan digunakan tiga buah metode integrasi, yaitu metode trapesium, metode Simpson 1/3 dan metode Romberg. Ketiga metode ini dipilih karena memiliki tingkat ketelitian yang berbeda-beda.

3.1. Metode Trapesium

Metode trapesium pada dasarnya menghitung nilai fungsi pada kedua batas integrasi, dan menghitung luas trapesium yang dihasilkan dari menghubungkan kedua titik batas. Akurasi metode trapesium dapat ditingkatkan dengan membagi rentang integrasi menjadi beberapa pias dan menerapkan metode trapesium pada setiap pias. Berikut ini adalah *pseudocode* untuk pemakaian metode trapesium dalam menghitung hasil integrasi:

```
Trapesium(bawah, atas, pias)

panjang ← (atas-bawah) / pias
x ← bawah
total ← (fungsi(atas) + fungsi(bawah)) /2
for I ← 1 to pias-1 do
        x ← x + panjang
        total ← total + fungsi(x)
luas ← total * panjang

→ luas
```

Hasil perhitungan metode trapesium memiliki tingkat ketelitian $O(h^2)$, yang berarti perbedaan hasil perhitungan dengan hasil integrasi sebenarnya berbanding lurus dengan kuadrat panjang rentang.

3.2. Metode Simpson 1/3

Metode Simpson 1/3 memiliki banyak persamaan dengan metode trapesium yang sudah dijelaskan sebelumnya, hanya saja perhitungan dilakukan untuk setiap pasang pias yang bersebelahan, dan daerah yang dihitung luasnya diasumsikan berbentuk kurva yang melewati ketiga titik pada pasangan pias tersebut. Kurva yang terbentuk pasti akan memenuhi persamaan kuadrat yang dapat ditentukan dari ketiga titik yang ada. Berikut ini adalah *pseudocode* dari metode Simpson 1/3:

```
Simpson1/3(bawah, atas, pias)

panjang ← (atas-bawah) / pias
x  ← bawah
total  ← fungsi(atas) + fungsi(bawah)
for  I  ← 1 to pias-1 do
        x ← x + panjang
        if (I mod 2 = 1) then
                total  ← total + 4 * fungsi(x)
        else
                total  ← total + 2 * fungsi(x)
luas  ← total / 3 * panjang

→ luas
```

Metode Simpson 1/3 memiliki tingkat ketelitian pada orde $O(h^4)$, walaupun memiliki kekurangan berupa semua titik harus berjarak sama (*equispaced*) dan jumlah pias harus kelipatan 2. Untuk jumlah pias ganjil metode Simpson 1/3 dapat dikombinasikan dengan metode Simpson 3/8. Dalam makalah ini metode Simpson 3/8 tidak digunakan karena memiliki tingkat ketelitian yang setara dengan metode Simpson 3/8.

3.3. Metode Romberg

Metode Romberg merupakan metode integrasi yang unik karena memiliki tingkat ketelitian yang ordenya dapat diatur sesuai dengan keperluan dan dalam proses perhitungannya melibatkan metode lain, biasanya metode trapesium karena kesederhanaan perhitungannya. Metode ini pada dasarnya membuat beberapa estimasi perhitungan untuk jumlah pias yang berbeda-beda, kemudian membuat perbaikan perhitungan berdasarkan estimasi-estimasi tersebut hingga didapat sebuah angka akhir. Jumlah pias yang dipakai adalah 2^i untuk setiap i dari 0 hingga n-1, dimana n adalah jumlah estimasi yang diinginkan. Tingkat ketelitian dipengaruhi oleh jumlah estimasi yang dipakai.

Berikut ini adalah *pseudocode* dari metode Romberg yang memakai metode trapesium didalamnya:

```
Romberg(bawah, atas, estimasi)

for  I ← 0 to estimasi-1 do
        hitung[I] ← trapesium(bawah,atas,2^I)

for  I ← 1 to estimasi do
        for j ← 0 to estimasi-I do
                hitung[j] = hitung[j+1]*4^I – hitung[j]
                hitung[j] = hitung[j] / (4^I - 1)

→ hitung[0]
```

Tingkat ketelitian dari metode ini berada pada orde $O(h^{2*estimasi})$, sehingga kita dalam teori dapat mencapai tingkat ketelitian sesuai keinginan kita dengan memakai metode ini. Dalam prakteknya, hal ini tidak selalu mungkin karena adanya beberapa keterbatasan.

4. Hasil Perhitungan dengan Metode Integrasi

Berikut ini diperlihatkan hasil-hasil perhitungan dari ketiga metode yang diberikan diatas. Semua perhitungan dilakukan dengan memakai metode bagi dua dengan tingkat akurasi yang diinginkan adalah 10^{-15} dan perhitungan dilakukan pada komputer Toshiba Portege Z935-P300 i5dengan sistem operasi Windows 7-64 bit. Program dibuat dalam bahasa C++ dengan memanfaatkan logika yang sama dengan *pseudocode* yang ditulis di atas dan memakai variabel dengan tingkat ketelitian double. Compiler yang dipakai adalah MinGW versi 4.7.2.

Tabel 1
Hasil Perhitungan Nilai *e* dengan Metode Integrasi

Metode	Jumlah Pias / Estimasi	Hasil Perhitungan (Kecocokan)
Trapesium	100000	2.718281828399848 (9)
Simpson 1/3	100000	2.718281828461860 (10)
Romberg	10	2.718281828459079 (13)

Yang dimaksud dengan Kecocokan pada tabel diatas adalah digit belakang koma terakhir pada hasil perhitungan yang masih sesuai dengan hasil perhitungan untuk 50 digit pertama belakang koma e, yang diambil dari OEIS[1].

Berdasarkan hasil perhitungan diatas, terlihat bahwa metode trapesium memiliki tingkat akurasi terendah, sebanding dengan tingkat ketelitian perhitungan yang diharapkan. Yang unik adalah hasil perhitungan metode Simpson 1/3 bisa melebihi estimasi nilai e, sedangkan kedua metode lainnya mendapatkan nilai yang lebih rendah dari estimasi nilai e. Alasan yang bisa diberikan penulis adalah ketiga metode ini hanya mengestimasi luas daerah integrasi, sehingga bisa jadi hasilnya lebih atau kurang dari luas seharusnya dan mempengaruhi proses perhitungan. Belum lagi adanya galat trunkasi untuk digit-

digit yang berada di bawah rentang ketelitian yang bisa disediakan komputer membuat hasil perhitungan menjadi lebih tidak teliti. Walaupun demikian, estimasi yang didapatkan diatas dapat dihitung dalam waktu kurang dari 1 detik, dan akurasi yang didapat biasanya sudah memadai untuk perhitungan biasa.

5. Cara Lain untuk menghitung *e*

Untuk menghitung nilai *e*, ada beberapa definisi lain yang dapat dimanfaatkan. Kedua definisi berikut adalah ekuivalen dan tepat bernilai *e* [3]:

a) $$\lim_{h \to \infty} (1 + 1/h)^h$$

b) $$\lim_{(n \to \infty)} \left(1 + \sum_{i=1}^{n} 1/i!\right)$$

Pseudocode program yang mengimplementasikan definisi yang ditulis diatas dapat dilihat pada 4 kotak berikut:

```
DefinisiA1(n)

jawab ← 1
kali ← 1 + 1 / n
for I ← 1 to n do
        jawab ← jawab * kali

→ jawab
```

```
DefinisiA2(n)

if (n == 0) return 1
if (n == 1) return kali
jawab ← DefinisiA2(n / 2)
jawab ← jawab * jawab
if (n mod 2 = 1) jawab ← jawab * kali
→ jawab

//Pada program utama
kali = 1 + 1 / n
→ DefinisiA2(n)
```

```
DefinisiB1(n)

jawab ← 0
tambah ← 1
for I ← 1 to n do
        tambah ← tambah * i
        jawab ← jawab + 1 / tambah

→ jawab + 1.0
```

```
DefinisiB2(n)

jawab ← 0
for I ← n downto 1 do
        jawab ← jawab + 1
        jawab ← jawab / I

→ jawab + 1.0
```

Program keempat didasarkan pada definisi B, hanya saja cara perhitungannya dilakukan dengan arah yang berkebalikan, dimulai dari n terbesar menuju n terkecil. Pada tiap iterasi berlaku persamaan jawab = 1/i + 1/(i*(i+1)) + 1/(i*(i+1)*(i+2)) + … + 1/(i*(i+1)*...*n). Ketika iterasi sudah selesai dilakukan maka hasil perhitungannya akan sesuai dengan definisi yang diberikan jika ditambah 1.

Definisi B disediakan dengan 2 cara yang berbeda untuk alasan yang akan dikemukakan pada Bab Selanjutnya.

6. Perbandingan antara Semua Cara

Berikut ini adalah tabel yang berisi hasil program yang menggunakan ketiga *pseudocode* yang diberikan diatas. Semua batasan yang berlaku pada ketiga program pertama juga berlaku pada ketiga program ini.

Tabel 2
Hasil Perhitungan Nilai *e* dengan Metode Lain

Metode	N	Hasil Perhitungan (Kecocokan)
Definisi A-1	800000	2.718280129431901 (5)
Definisi A-2	800000	2.718280129483476 (5)
Definisi B-1	20	2.718281828459046 (14)
Definisi B-2	20	2.718281828459045 (15)

Dari tabel 1 dan 2, terlihat bahwa menggunakan definisi A untuk mencari nilai *e* bukan ide yang baik, karena dengan memanfaatkan nilai N yang sudah sangat besar sekalipun hasilnya masih lebih rendah bahkan dari estimasi menggunakan trapesium. Memang ada kemungkinan hal ini terjadi karena keterbatasan komputer sehingga banyak digit hasil perkalian yang terbuang pada setiap tahap perkalian sehingga hasilnya menjadi sangat tidak akurat. Hanya saja hasil dari Definisi A-2 tidak lebih baik dari Definisi A-1, padahal jumlah perkalian yang dilakukan sudah lebih sedikit sehingga seharusnya mengurangi jumlah digit yang terbuang selama proses perhitungan. Maka dapat disimpulkan penggunaan definisi A yang merupakan definisi pertama dari *e* merupakan pilihan yang buruk.

Hasil yang berbanding terbalik didapatkan pada pengunaan definisi B, baik untuk B-1 maupun B-2. Dengan penggunaan N sekecil 20 saja, didapatkan akurasi yang sebesar 14 dan 15 digit belakang koma. Hal ini menunjukkan bahwa untuk mencari estimasi *e* yang seakurat mungkin dalam waktu singkat, sebaiknya cara yang digunakan adalah definisi B-1 atau B-2. Cara B-2 memggunakan 1 penjumlahan dan 1 pembagian untuk setiap iterasi, sedangkan cara B-1 menggunakan 2 pembagian/perkalian dan 1 penjumlahan untuk setiap iterasi. Maka secara kompleksitas waktu, cara B-2 unggul dibandingkan dengan cara B-1.

7. Kesimpulan

Penggunaan metode numerik integrasi untuk mencari estimasi nilai e yang akurat hingga digit tertentu dapat memberikan hasil yang cukup baik untuk perhitungan biasa, dengan metode Romberg memberikan hasil yang paling akurat dibandingkan metode lainnya. Akurasi dapat ditingkatkan dengan menggunakan jumlah estimasi yang lebih banyak serta memakai komputer yang memiliki presisi yang lebih tinggi.

Untuk mencari estimasi e yang seakurat mungkin, cara B-1 dan B-2 dapat digunakan sebagai pilihan utama. Hanya saja, dalam prakteknya cara B-1 memiliki kelemahan berupa nilai yang sangat kecil dapat tidak tercatat dalam hasil akhir karena komputer sudah tidak sanggup menyimpannya, padahal bisa jadi kumpulan nilai kecil ini jika dijumlahkan hasilnya menjadi signifikan. Hal ini tidak tampak pada hasil karena perbandingan nilai hasil pembagian pada setiap tahap sangatlah besar, sehingga nilai-nilai yang kecil dengan cepat menjadi tidak signifikan. Tetap saja, jika tujuannya adalah mencari estimasi yang akurat dengan secepat mungkin, cara B-2 sebaiknya dijadikan pilihan utama.

8. Referensi

[1] http://oeis.org/A001113 , diakses tanggal 5 Mei 2013.
[2] http://www-history.mcs.st-and.ac.uk/HistTopics/e.html , diakses tanggal 5 Mei 2013.
[3] Dale Varberg, Edwin J. Purcell, Steven E. Rigdon; *Calculus: Ninth Edition*; 2007, Pearson Education.
[4] http://mathworld.wolfram.com/e.html , diakses tanggal 5 Mei 2013.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 5 Mei 2013

Christianto - 13510003