Menghitung Ketinggian Rata-Rata Pohon Terurut

Archie Anugrah - 13508001

Jurusan Teknik Informatika Institut Teknologi Bandung
Jalan Ganesha nomor 10, Bandung
e-mail: if18001@students.if.itb.ac.id

ABSTRAK

Dalam makalah ini, penulis membahas tentang salah satu teorema yang tergolong baru dalam teori graf pohon, yaitu teorema fungsi perhitungan ketinggian rata-rata dari pohon terurut. Teorema yang diturunkan oleh N.G. de Brujin, D.E. Knuth, dan S.O. Rice ini menggunakan metoda perhitungan jumlah deret, perhitungan bilangan kompleks, dan aproksimasi Stirling untuk menurunkan teorema fungsi penghitungan ketinggian rata-rata dari pohon terurut. Walaupun penurunan teorema menggunakan aproksimasi, galat yang dihasilkan tergolong kecil (sebanding dengan log n) sehingga dapat diabaikan. Ketinggian rata-rata dari pohon terurut dibutuhkan ketika kita mengaproksimasi kebutuhan memori untuk program yang kita buat, terutama pemrosesan pohon dengan struktur data *stack* (tumpukan) dan *queue* (barisan). Aproksimasi kebutuhan penggunaan memori penting untuk menentukan performa dari program yang kita buat (*benchmark*). Dengan teorema ini, penghitungan ketinggian rata-rata dari pohon dengan simpul tertentu tidak perlu dicari dengan mencoba dan menghitung semua kemungkinan pohon satu per satu yang tentunya akan menghemat waktu yang dibutuhkan dalam pencarian ketinggian rata-rata dari pohon terurut.

Kata kunci: Fungsi perhitungan ketinggian rata-rata pohon terurut, penurunan teorema, penggunaan perhitungan ketinggian rata-rata pohon terurut.

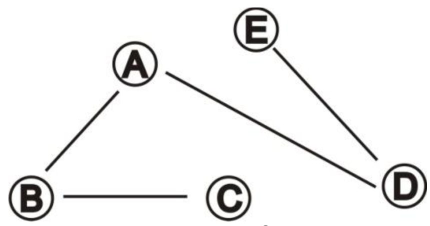
1. PENDAHULUAN

1.1 Definisi Graf

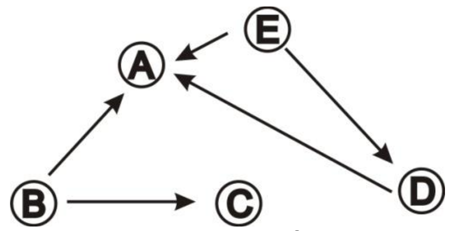
Graf adalah pasangan himpunan (V,E) di mana V adalah himpunan tidak kosong dari vertex(simpul) dan E adalah himpunan sisi yang menghubungkan sepasang simpul dalam graf tersebut.

Graf, menurut himpunan sisinya, dibagi menjadi :

1. Graf tak-berarah (*undirected graph*)
   Graf yang sisinya tidak mempunyai orientasi arah.
2. Graf berarah (*directed graph atau digraph*)
   Graf yang sisinya mempunyai orientasi arah.



Gambar 1.Contoh Graf Tak Berarah

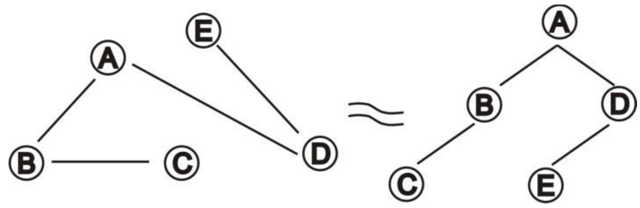


Gambar 2.Contoh Graf Berarah

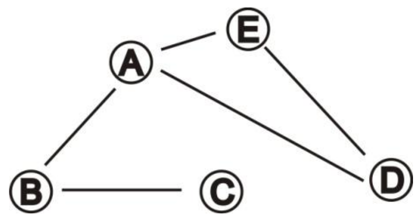
1.2 Pohon

1.2.1 Definisi Pohon

Pohon didefinisikan sebagai graf tak-berarah terhubung yang tidak mengandung sirkuit. Jadi pohon sebenarnya hanyalah salah satu spesifikasi dari graf.



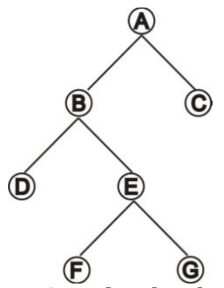
Gambar 3.Contoh Graf Pohon



Gambar 4.Contoh Graf Bukan Pohon

1.2.2 Definisi Pohon Berakar

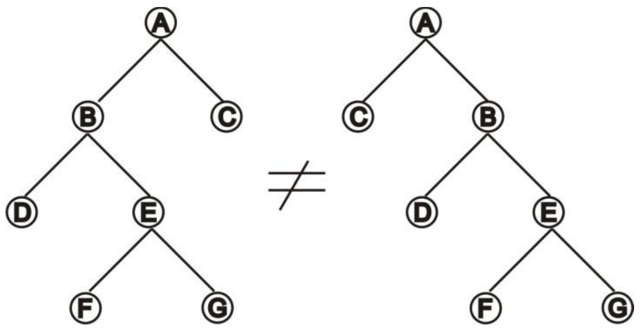
Pohon berakar (*rooted tree*) adalah pohon yang satu buah simpulnya diperlakukan sebagai akar dan sisi-sisinya diberi arah sehingga menjadi graf berarah. Menurut perjanjian, dalam melukiskan pohon berakar, kita tidak perlu menggambarkan panah dari graf berarah.



Gambar 5.Contoh Pohon berakar

1.2.3 Definisi Pohon Terurut

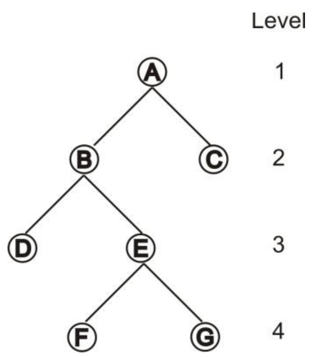
Pohon terurut (*ordered tree*) adalah pohon berakar yang urutan anak-anaknya penting.



Gambar 6.Keterurutan Pohon

1.2.4 Definisi Tinggi Pohon

Level (aras) adalah tingkatan dari pohon dimana penomoran level bermula dari akar yang diberi nilai 1.



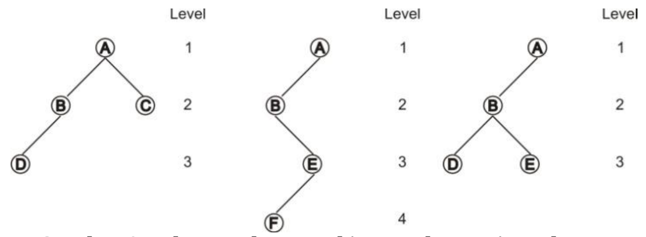
Gambar 7.Level Pohon

Ketinggian adalah aras paling maksimal dari suatu pohon tertentu. Contoh : Pohon pada gambar 7 memiliki ketinggian 4.

1.3 Ketinggian Rata-rata Pohon Terurut

Rumus ketinggian rata-rata dari pohon terurut adalah rumus yang dikembangkan oleh N.G. de Brujin, D.E.Knuth, dan S.O.Rice untuk menghitung rata-rata ketinggian dari pohon terurut dengan jumlah simpul tertentu.

Misalkan, kita diberikan 4 buah simpul yang akan dibuat menjadi pohon maka kita dapat membuat beberapa tipe pohon, seperti :



Gambar 8.Beberapa kemungkinan pohon 4 simpul

Di contoh diatas, baru disebutkan contoh-contoh pohon biner saja, belum lagi untuk pohon n-ary lainnya sehingga akan terbentuk begitu banyak kemungkinan pohon dengan ketinggian yang berbeda-beda untuk jumlah simpul yang tertentu.

Untuk menghitung rata-rata ketinggian dari pohon, biasanya digunakan pendekatan *brute force*. Pendekatan ini dikerjakan dengan mengkonstruksi semua pohon yang mungkin dari jumlah simpul yang ada. Setelah semua pohon terbentuk, ketinggian masing-masing pohon dihitung, dijumlah, lalu dirata-rata. Metoda ini memang dapat menghasilkan nilai rata-rata yang bisa dibilang akurat, tetapi algoritma ini membutuhkan waktu yang lama dan tidak efisien karena sifat dari fungsi-fungsi yang menggunakan pendekatan *brute force* memiliki kompleksitas waktu eksponensial.

Kita butuh menghitung ketinggian pohon untuk menghitung jumlah memori yang dibutuhkan untuk memroses pohon tertentu (misalkan algorima memroses pohon ekspresi yang harus mengkonversi pohon biner kedalam *stack* terlebih dahulu). Ketinggian rata-rata dapat digunakan untuk memrediksi berapa jumlah memori yang kira-kira diperlukan untuk pohon dengan simpul tertentu. Jadi terlihat bahwa kebutuhan pengihtungan ketinggian pohon rata-rata ini berhubungan erat dengan pengujian ketahanan algoritma tertentu dalam menangani data pohon, sehingga tidaklah efektif bila kita harus menggunakan *brute force* dalam pencarian ketinggian pohon terurut rata-rata.

2. METODE

2.1 Penurunan Rumus

Misalkan ketinggian pohon disimbolkan sebagai $A_h(z)$.

$$A_h(z) = \sum A_{nh} z^n \qquad (1)$$

Kita dapat mencari semua pohon dengan tinggi ≤ h+1 dengan menghilangkan akar dan menambahkan upapohon dengan tinggi ≤ h, sehingga :

$$\begin{aligned} A_{h+1}(z) &= z(1 + A_h(z) + A_h(z)^2 + A_h(z)^3 + \cdots) \\ &= z/(1 - A_h(z)), \qquad h \geqslant 0. \end{aligned} \qquad (2)$$

Karena, $A_0(z) = 0$, maka rumus (2) dapat diubah menjadi rumus rekurens untuk A_{nh}

$$\begin{aligned} A_{n,h+1} = A_{n-1,h+1}A_{1,h} + A_{n-2,h+1}A_{2,h} + \cdots + A_{1,h+1}A_{n-1,h}, \\ n \geqslant 2, \quad h \geqslant 0, \end{aligned} \qquad (3)$$

Karena tidak ada pohon dengan ketinggian > n, maka kita mendapatkan rumusan (rumus ini sebenarnya merupakan rumus total pohon dengan n simpul):

$$A_{nh} = A_{nn} = \binom{2n-2}{n-1}\frac{1}{n}, \qquad h \geqslant n, \qquad (4)$$

Jika kita menghitung iterasi (4), maka kita akan mendapat pembagian sebagai berikut :

$$A_4(z) = \cfrac{z}{1 - \cfrac{z}{1 - \cfrac{z}{1 - z}}}. \qquad (5)$$

Fungsi diatas dapat ditulis dalam bentuk :

$$A_h(z) = zp_h(z)/p_{h+1}(z), \qquad (6)$$

Dimana

$$p_0(z) = 0, \quad p_1(z) = 1, \quad p_{h+1}(z) = p_h(z) - zp_{h-1}(z). \qquad (7)$$

Solusi untuk fungsi rekurens tersebut adalah

$$p_h(z) = (1-4z)^{-1/2}\left(\left(\frac{1+(1-4z)^{1/2}}{2}\right)^h - \left(\frac{1-(1-4z)^{1/2}}{2}\right)^h\right), \qquad (8)$$

Jika kita menggunakan $z = \frac{1}{4\,(\cos\theta)^2}$, kita akan mendapatkan :

$$\begin{aligned} p_h(4\cos^2\theta)^{-1} &= \sin h\theta/(\sin\theta(2\cos\theta)^{h-1}), \\ A_h(4\cos^2\theta)^{-1} &= \sin h\theta/(2\cos\theta\sin(h+1)\theta). \end{aligned} \qquad (9)$$

Jika kita tilik lebih jauh kita akan mendapatkan bahwa $p_h(-1)$ adalah bilangan Fibonacci F_h dan

$$p_h(z) = \sum_{0 \leqslant k < h} \binom{h-1-k}{k}(-z)^k, \qquad h \geqslant 1. \qquad (10)$$

Karena $ph(z)^2 - p_{h+1}(z)\,p_{h-1}(z) = z^{h-1}$, maka didapatkan :

$$A_h(z) - A_{h-1}(z) = z^h/p_{h+1}(z)p_h(z). \qquad (11)$$

Karena ph adalah fungsi polinom berderajat $\frac{h-1}{2}$, akar $p_h(z)=0$ adalah $(4\cos^2(\frac{j\pi}{h}))^{-1}$, didapatkan ekspansi parsial sebagai berikut :

$$A_h(z) = \sum_{1 \leqslant j \leqslant h/2} \frac{\tan^2\theta_{jh}}{(h+1)(1-(4\cos^2\theta_{jh})z)} + a_h + b_h z, \qquad (12)$$

Dengan $\theta_{jh} = \frac{j\pi}{h+1}$ dan

$$\begin{aligned} a_{2m} &= -m, & b_{2m} &= 0, \\ a_{2m+1} &= -m(2m+1)/6(m+1), & b_{2m+1} &= (m+1)^{-1}, \quad m \geqslant 1. \end{aligned} \qquad (13)$$

Semua persamaan tersebut akan menghasilkan persamaan :

$$A_{nh} = (h+1)^{-1}\sum_{1 \leqslant j \leqslant h/2} 4^n \sin^2(j\pi/(h+1))\cos^{2n-2}(j\pi/(h+1)), \qquad n \geqslant 2. \quad (14)$$

Persamaan (14) akan menghasilkan nilai yang konstan untuk n tertentu dan h ≥ n. Kasus khusus dari (14) dapat diturunkan menjadi formula asimtotik berikut

$$A_{nh} \sim (4^n/(h+1))\tan^2(\pi/(h+1))\cos^{2n}(\pi/(h+1)), \qquad \text{fixed } h, \quad n \to \infty. \quad (15)$$

Penurunan A_{nh} juga dapat diturunkan dari teori variable kompleks sebagai berikut :

$$\begin{aligned} A_{nh} &= (2\pi i)^{-1}\int^{(0+)} \frac{dz}{z^{n+1}} A_h(z) \\ &= (2\pi i)^{-1}\int^{(0+)} \frac{dz}{z^n}(1+u)\frac{1-u^h}{1-u^{h+1}}, \end{aligned} \qquad (16)$$

dimana

$$u = \frac{1-(1-4z)^{1/2}}{1+(1-4z)^{1/2}}, \qquad (17)$$

Karena

$$z = u/(1+u)^2, \qquad (18)$$

Kita mendapatkan bahwa $u \approx z$ ketika $|z| < 1$, sehingga kita dapat mengubah variable (16) untuk mendapat persamaan berikut

$$A_{nh} = (2\pi i)^{-1}\int^{(0+)}\frac{du}{u^n}(1-u)(1+u)^{2n-2}\frac{1-u^h}{1-u^{h+1}}. \qquad (19)$$

Dengan kata lain A_{nh} adalah koefisien dari u^{n-1} dalam $(1-u)(1+u)2n-2$ $\frac{1-u^h}{1-u^{h-1}}$. Untuk menyederhanakan fungsi, kita akan memfokuskan pada pohon dengan ketinggian lebih besar dari h.

$$\begin{aligned} B_{nh} &= A_{nn} - A_{nh} \\ &= (2\pi i)^{-1}\int^{(0+)}\frac{du}{u^{n+1}}(1-u)^2(1+u)^{2n-2}\frac{u^{h+1}}{1-u^{h+1}}. \end{aligned} \qquad (20)$$

Sehingga

$$B_{n+1,h-1} = \sum_{k\geq 1}\left(\binom{2n}{n+1-kh} - 2\binom{2n}{n-kh} + \binom{2n}{n-1-kh}\right). \qquad (21)$$

Tinggi rata-rata pohon dengan node $\frac{S_n}{A_{nn}}$ dimana S_n adalah jumlah dari :

$$\begin{aligned} S_n &= \sum_{h\geq 1} h(A_{nh} - A_{n,h-1}) \\ &= \sum_{h\geq 1} h(B_{n,h-1} - B_{nh}) \\ &= \sum_{h\geq 0} B_{nh} \\ &= (2\pi i)^{-1}\int^{(0+)}\frac{du}{u^{n+1}}(1-u)^2(1+u)^{2n-2}\sum_{h\geq 1}\frac{u^h}{1-u^h} \\ &= (2\pi i^{-1})\int^{(0+)}\frac{du}{u^{n+1}}(1-u)^2(1+u)^{2n-2}\sum_{k\geq 1} d(k)u^k. \end{aligned} \qquad (22)$$

Dimana $d(k)$ menyimbolkan jumlah pembagi positif dari k.

$$S_{n+1} = \sum_{k\geq 1} d(k)\left(\binom{2n}{n+1-k} - 2\binom{2n}{n-k} + \binom{2n}{n-1-k}\right). \qquad (23)$$

Sekarang kita harus mencari fungsi asimptotik dari fungsi berikut :

$$f_a(n) = \sum_{k\geq 1}\left(\binom{2n}{n+a-k}\bigg/\binom{2n}{n}\right)d(k), \quad a \text{ konstan } n\to\infty, \qquad (24)$$

Dengan menggunakan $x = \frac{k-1}{n}$, dengan aproksimasi Stirling kita dapat mendapatkan

$$\begin{aligned} \binom{2n}{n+a-k}\bigg/\binom{2n}{n} &= \exp\left(-2n\left(\frac{x^2}{1\cdot 2}+\frac{x^4}{3\cdot 4}+\cdots\right)+\left(\frac{x^2}{2}+\frac{x^4}{4}+\cdots\right)\right. \\ &\quad \left. -\frac{1}{6n}(x^2+x^4+\cdots)+O(x^2n^{-3})\right), \end{aligned} \qquad (25)$$

Saat $-\frac{1}{2} < x < \frac{1}{2}$, dan

$$\binom{2n}{n+a-k}\bigg/\binom{2n}{n} = O(\exp(-n^{2\varepsilon}))$$

Untuk $k \geq n^{\frac{1}{2}+\varepsilon} + a$ untuk $\varepsilon > 0$. Oleh karena itu jumlah deret untuk $k \geq n^{\frac{1}{2}+\varepsilon} + a$ dapat diabaikan.

Sekarang kita beralih ke fungsi berikut :

$$g_b(n) = \sum_{k\geq 1} k^b d(k)\exp(-k^2/n), \qquad \text{fixed } b, \qquad n\to\infty. \qquad (26)$$

Lagi-lagi untuk $k=n^{\frac{1}{2}+\varepsilon}$ jadi kita bisa menggunakan persamaan (25) untuk mengekspresikan f dalam g.

$$\begin{aligned} f_a(n) &= g_0(n) + \frac{2a}{n}g_1(n) - \frac{a^2}{n}g_0(n) + \frac{4a^2+1}{2n^2}g_2(n) - \frac{1}{6n^3}g_4(n) \\ &\quad - \frac{2a^3+a}{n^2}g_1(n) + \frac{4a^3+5a}{3n^3}g_3(n) - \frac{a}{3n^4}g_5(n) + O(n^{-2+\varepsilon}g_0(n)). \end{aligned} \qquad (27)$$

Perilaku dari g_b dapat diturunkan dengan menggunakan formula

$$e^{-x} = (2\pi i)^{-1}\int_{c-i\infty}^{c+i\infty}\Gamma(z)x^{-z}dz, \qquad c>0, \quad x>1, \qquad (28)$$

$$\begin{aligned} g_b(n) &= \sum_{k\geq 1}(2\pi i)^{-1}\int_{c-i\infty}^{c+i\infty} n^z\Gamma(z)k^{b-2z}d(k)\,dz \\ &= (2\pi i)^{-1}\int_{c-i\infty}^{c+i\infty} n^z\Gamma(z)\zeta(2z-b)^2dz, \end{aligned} \qquad (29)$$

$$n^{\frac{1}{2}(b+1)}\Gamma(\tfrac{1}{2}(b+1))(\tfrac{1}{4}\ln n + \tfrac{1}{4}\psi(\tfrac{1}{2}(b+1)) + \gamma). \qquad (30)$$

Residue saat $z=-k$ adalah

$$n^{-k}(-1)^k\zeta(-2k-b)^2/k! = n^{-k}(-1)^k B^2_{2k+b+1}/(2k+b+1)^2k! \qquad (31)$$

Yang selalu hampir bernilai 0 saat b genap. Jumlah (30) dan (31) untuk semua $k \geq 0$ memberikan baris asimptotik dari $g_b(n)$. Sehingga untuk semua $m>0$

$$\begin{aligned} g_0(n) &= \tfrac{1}{2}(\pi n)^{1/2}\ln n + (\tfrac{3}{4}\gamma - \tfrac{1}{2}\ln 2)(\pi n)^{1/2} + \tfrac{1}{4} + O(n^{-m}); \\ g_1(n) &= \tfrac{1}{2}n\ln n + \tfrac{1}{2}\gamma n + (\tfrac{1}{144}) - (\tfrac{1}{14400})n^{-1} + O(n^{-2}); \\ g_2(n) &= (n/8)(\pi n)^{1/2}\ln n + (\tfrac{1}{4} + \tfrac{3}{8}\gamma - \tfrac{1}{4}\ln 2)n(\pi n)^{1/2} + O(n^{-m}); \end{aligned} \qquad (32)$$

Kembali ke masalah semula, kita memunyai

$$
\begin{aligned}
S_{n+1}/(n+1)A_{n+1,n+1} &= f_1(n) - 2f_0(n) + f_{-1}(n) \\
&= (-2/n)g_0(n) + (4/n^2)g_2(n) + O(n^{-5/2}\log n)
\end{aligned} \tag{33}
$$

Menggunakan persamaan (4), (23), (24), (27) yang sebanding dengan.

$$\pi n^{\frac{1}{2}} - \frac{1}{2} + O\left(n^{-\frac{1}{2}} \log n\right) \tag{34}$$

Jadi, teorema yang didapatkan adalah untuk pohon terurut bersimpul n, didapatkan bahwa tinggi rata-rata sebesar

$$\pi n^{\frac{1}{2}} - \frac{1}{2} + O\left(n^{-\frac{1}{2}} \log n\right)$$

Dimana faktor error sebanding log n yang dapat dibilang kecil.

2.2. Penggunaan

Dengan teorema fungsi tinggi rata-rata pohon terurut, kita dapat memrediksikan rata-rata tinggi pohon terurut tanpa menggunakan algoritma yang mencoba semua kemungkinan pohon yang dapat terbentuk. Dengan adanya teorema yang berbentuk fungsi tersebut, lama eksekusi menurun dengan sangat drastis, dari yang sebelumnya menggunakan traversal elemen satu per satu, menjadi sekedar pemanggilan fungsi belaka.

Jika dilihat dari penurunan rumus memang, dalam penghitungan digunakan metoda aproksimasi, tetapi galat yang dihasilkan hanya sebanding dengan log n yang menyebabkan galat yang terjadi dapat dibilang kecil dan dapat diabaikan.

3. KESIMPULAN

Penurunan fungsi untuk nilai rata-rata tinggi pohon terurut membuat pencarian nilai rata-rata tinggi pohon terurut menjadi efektif karena untuk mencari nilai rata-rata tinggi pohon terurut tidak perlu lagi dicari semua kombinasi pohon satu per satu.

REFERENSI

[1] Munir, Rinaldi. 2008. Diktat Kuliah IF2091 Struktur Diskrit, edisi keempat.Program Studi Teknik Informatika, Sekolah Teknik Elektro dan Informatika, Institut Teknologi Bandung.

[2] Read, Ronald C, “Graph Theory And Computing”, Academic Press, 1972.

[3] http://stackoverflow.com/questions/861393/average-height-of-a-binary-search-tree (Waktu akses : 11 Desember 2009, 7:05 PM)

[4] http://en.wikipedia.org/wiki/Tree_%28graph_theory%29 (Waktu akses : 11 Desember 2009, 7:05 PM)