

USULAN TUGAS AKHIR

1. IDENTITAS PENGUSUL

NAMA : BAMBANG DWI PRASETYO RAMADHAN
NRP : 5110100053
DOSEN WALI : Dr. Ir. Siti Rochimah, M.T.
DOSEN PEMBIMBING : 1. Dwi Sunaryono, S.Kom., M.Kom.
2. Adhatus Solichah, S.Kom., M.Sc.

2. JUDUL TUGAS AKHIR

“Perancangan dan Pembangunan Perangkat Lunak Rekomendasi Lagu Pada Karaoke Player Berdasarkan Jenis Suara”

3. LATAR BELAKANG

Karaoke merupakan sebuah hiburan dalam bentuk seseorang menyanyi diiringi dengan musik dan teks lirik yang ditunjukkan pada sebuah layar. Karaoke sangat populer di kalangan masyarakat karena selain dapat menghilangkan stress, karaoke juga seru untuk dilakukan bersama teman. Aplikasi karaoke yang ada saat ini masih belum memiliki fitur yang dapat memudahkan penggunaanya dalam memilih lagu yang sesuai dengan kemampuan pengguna, sering sekali pengguna harus bersusah payah dalam menyanyikan sebuah lagu karena nada pada lagu tersebut terlalu tinggi atau terlalu rendah bagi pengguna. Banyak orang belum mengetahui secara pasti jenis suara yang dimilikinya maka dari itu terkadang mereka memilih lagu yang tidak sesuai dengan jenis suaranya sehingga yang terjadi nada yang keluar tidak sesuai dengan nada yang seharusnya, terkadang nada yang dihasilkan lebih rendah atau lebih tinggi. Setiap orang memiliki batas terendah dan tertinggi dari nada yang dapat dicapainya.

Jenis suara merupakan rentangan nada yang dimiliki oleh setiap orang. Setiap orang bisa memiliki jenis suara yang sama atau berbeda dengan orang lain. Jenis suara ada 6 macam di antaranya adalah tenor, bariton, bass (pria) dan sopran, *mezzo-sopran*,

alto (wanita) [1]. Beberapa pria terkadang dapat mencapai jenis suara yang dimiliki wanita begitu juga sebaliknya.

4. RUMUSAN MASALAH

Rumusan masalah yang akan diangkat dalam Tugas Akhir ini dapat dipaparkan sebagai berikut:

1. Bagaimana cara membuat aplikasi karaoke berbasis desktop yang dapat memberikan rekomendasi lagu?
2. Bagaimana cara mengolah data suara pengguna untuk dapat dibedakan berdasarkan jenis suara?
3. Bagaimana cara memberikan rekomendasi lagu berdasarkan jenis suara pengguna?

5. BATASAN MASALAH

Permasalahan yang dibahas dalam Tugas Akhir ini memiliki beberapa batasan masalah, di antaranya sebagai berikut:

1. Aplikasi dapat memberikan rekomendasi lagu berdasarkan jenis suara.
2. Tidak membedakan suara berdasarkan jenis kelamin.
3. Bahasa pemrograman yang digunakan adalah C# dengan *framework* .NET 4.0.

6. TUJUAN PEMBUATAN TUGAS AKHIR

Tujuan dari Tugas Akhir ini adalah sebagai berikut:

1. Memberikan rekomendasi lagu berdasarkan jenis suara.
2. Memberikan informasi jenis suara yang dimiliki oleh pengguna.

7. MANFAAT TUGAS AKHIR

Dengan adanya aplikasi ini dapat membantu seseorang dalam menemukan lagu yang memang cocok untuk dinyanyikan berdasarkan jenis suaranya sehingga orang tersebut dapat bernyanyi pada nada yang memang sesuai dengan kemampuannya.

8. TINJAUAN PUSTAKA

1. Analisis Frekuensi

Salah satu algoritma untuk pengolahan sinyal adalah *Fast Fourier Transform* yang merupakan algoritma untuk menghitung DFT (*Discrete Fourier Transform*) dengan cepat. FFT (*Fast Fourier Transform*) dapat digunakan untuk pengolahan sinyal digital.

DFT berperan penting untuk implementasi algoritma dalam pengolahan sinyal. *Fast Fourier Transform* merupakan proses lanjutan dari DFT. Transformasi ini dilakukan untuk mentransformasikan sinyal dari domain waktu ke domain frekuensi. Hal ini bertujuan agar sinyal dapat diproses dalam *spectral subtraction*. Hasil transformasi ini dipengaruhi oleh beberapa parameter, yaitu *sample rate* sinyal suara dan ukuran FFT.

Sample rate adalah banyaknya *sample input* analog yang diambil secara digital. *Sample rate* sinyal suara berpengaruh pada besarnya jangkauan frekuensi dari koefisien hasil FFT. Jangkauan frekuensi hasil FFT adalah setengah dari *sample rate* sinyal suara yang ditransformasi. Artinya, apabila terdapat sinyal suara dengan *sample rate* 44100 Hz, maka koefisien-koefisien hasil transformasi dari sinyal suara tersebut berkisar dari 0 Hz sampai 22050 Hz. Jadi, semakin besar *sample rate*, maka akan semakin detail pula *sample* analog yang diambil secara digital [2]. *Discrete Fourier Transform* didefinisikan oleh Persamaan 1.

$$S(m) = \sum_{n=0}^{N-1} x(n) e^{-\frac{i2\pi nm}{N}}, m=0,1,2,\dots,N-1 \quad (1)$$

Nilai $S(m)$ adalah nilai spektrum pada frekuensi. Penghitungan $f_{analysis(m)}$ menggunakan Persamaan 2.

$$f_{analysis(m)} = \frac{m f_s}{N} \quad (2)$$

Pada Persamaan 2, f_s menunjukkan frekuensi yang digunakan untuk *sample* sinyal *input*. Untuk menghitung *power spectrum* dan merepresentasikannya ke dalam *normalized decibel scale* menggunakan Persamaan 3.

$$S_{power}(m) = 20 \log_{10} \left(\frac{|S(m)|}{|S(m_{max})|} \right) dB \quad (3)$$

Pada Persamaan 3, m_{max} adalah indeks dari nilai DFT dengan nilai *power* tertinggi.

2. Analisis Spektrum

Hasil dari analisis frekuensi diperoleh informasi *harmonic content* dari sinyal *input*. Berdasarkan informasi tersebut dapat dicari nadanya sesuai dengan *harmonic structure* yang diterima [2]. Maka dari itu diperlukan penghitungan nilai diskrit spektrum menggunakan Persamaan 4.

$$S = S(x) = \sum_{n=0}^{N-1} S_n \delta(x - n) = \sum_{n=0}^{N-1} S_n \delta_n \quad (4)$$

N adalah jumlah dari frekuensi *bands*, n adalah indeks dari frekuensi *bands* dan δn adalah fungsi *Dirac delta*. Untuk menghitung *Boolean spectrum* menggunakan Persamaan 5.

$$S = \bigvee S(n) \delta_n, \quad s(n) = \begin{cases} 0 & \text{if } S_n = 0, \\ 1 & \text{if } S_n \neq 0, \end{cases} \quad (5)$$

Untuk mengubah *power spectrum* ke dalam bentuk diskrit spektrum dengan *simultaneous rescaling* dari sumbu frekuensi menggunakan Persamaan 6.

$$n = \left\lceil C \log_2 \frac{f}{f_0} + 0,5 \right\rceil \quad (6)$$

C adalah bilangan konstan yang sama dengan frekuensi *bands per octave*, f_0 adalah nilai tengah dari frekuensi nada dengan indeks 0.

9. RINGKASAN ISI TUGAS AKHIR

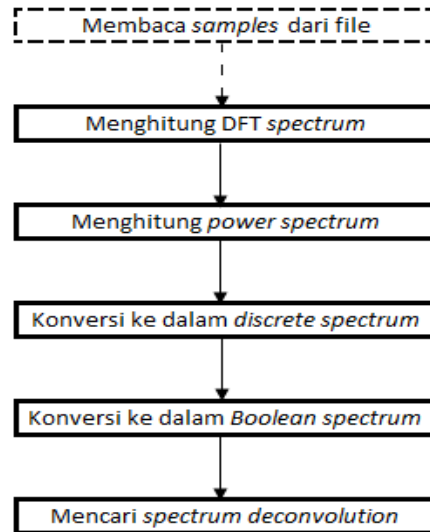
Penyanyi pada dasarnya memiliki jenis suara yang berbeda-beda. Jenis suara pada umumnya dibedakan menjadi 6 jenis suara yaitu tenor, bariton, bass, sopran, *mezzo-sopran* dan alto [1]. Setiap jenis suara memiliki rentangan nada yang berbeda, jenis suara dan rentangan nadanya dapat dipaparkan sebagai berikut:

- Tenor (C_3 - C_5).
Tenor merupakan suara tinggi dari pria.
- Bariton (F_2 - F_4)
Bariton merupakan suara tengah pria.
- Bass (E_2 - E_4)
Bass merupakan suara rendah pria.
- Sopran (C_4 - C_6)
Sopran merupakan suara tinggi wanita.
- *Mezzo-Sopran* (A_3 - A_5)
Mezzo-Sopran merupakan suara tengah wanita.
- Alto (F_3 - F_5)
Alto merupakan suara rendah wanita.

Aplikasi yang akan dibuat pada Tugas Akhir ini memiliki fitur-fitur dasar karaoke seperti pilih lagu, mainkan lagu, berhenti, jeda dan putar selanjutnya. Selain fitur dasar karaoke yang dimiliki oleh aplikasi ini, aplikasi ini juga memiliki fitur untuk memberikan rekomendasi lagu berdasarkan jenis suara.

Fase 1

Pada fase ini sistem akan melakukan *Chord Recognition*, *Chord Recognition* merupakan proses untuk mendapatkan atau mengenali nada pada suara [2]. *Chord Recognition* memiliki beberapa tahapan yang secara garis besar digambarkan pada Gambar 1.



Gambar 1. Tahapan Chord Recognition

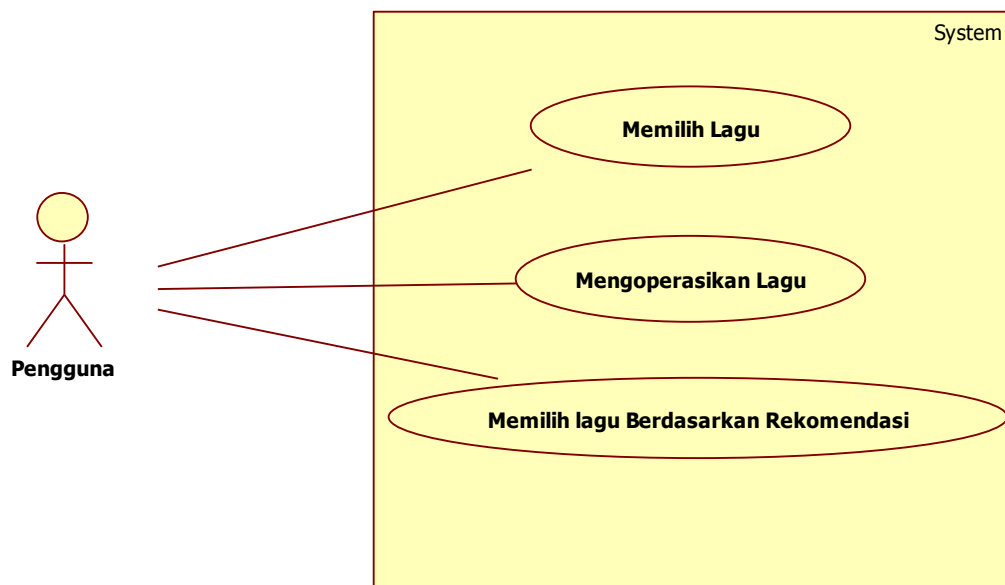
Tahapan yang pertama sistem akan merekam suara pengguna dan menyimpannya dengan format *Waveform Audio File Format* dan melakukan proses analisis frekuensi. Hasil rekaman tersebut akan diolah menggunakan metode FFT (*Fast Fourier Transform*). Perhitungan dari N -point DFT dapat menggunakan Persamaan 1 yang membutuhkan perkalian kompleks dari $O(N)$, sehingga penggunaan algoritma FFT diperlukan untuk mengurangi tingkat kompleks perhitungan menjadi $O(N \log 2N)$. Tahapan selanjutnya adalah dengan menghitung *power spectrum* dan mendefinisikannya ke *normalized decibel scale* menggunakan Persamaan 3. Tahapan selanjutnya *power spectrum* tersebut dikonversi ke dalam bentuk diskrit spektrum menggunakan Persamaan 6, selanjutnya diskrit spektrum dari *signal power* dikonversi menjadi *Boolean spectrum* menggunakan Persamaan 5. Tahapan terakhir adalah dengan mencari *deconvolution* $t * i + \varepsilon - \lambda$ dari *Boolean spectrum*, yang mana *interval distribution* i yang kita cari adalah hasil dari nomor dan *pitch* dari nada pada *chord* [2]. Tabel 1 adalah frekuensi dari masing-masing nada.

Tabel 1. Nada dan Frekuensinya (Hz) [2]

C_2	65.41	C_3	130.81	C_4	261.63	C_5	523.25
D_2^b/C_2^{\sharp}	69.30	D_3^b/C_3^{\sharp}	138.59	D_4^b/C_4^{\sharp}	277.18	D_5^b/C_5^{\sharp}	554.37
D_2	73.42	D_3	146.83	D_4	293.66	D_5	587.33
E_2^b/E_2^{\sharp}	77.78	E_3^b/E_3^{\sharp}	155.56	E_4^b/E_4^{\sharp}	311.13	E_5^b/E_5^{\sharp}	622.25
E_2	82.41	E_3	164.81	E_4	329.63	E_5	659.26
F_2	87.31	F_3	174.61	F_4	349.23	F_5	698.46
G_2^b/F_2^{\sharp}	92.50	G_3^b/F_3^{\sharp}	185.00	G_4^b/F_4^{\sharp}	369.99	G_5^b/F_5^{\sharp}	739.99
G_2	98.00	G_3	196.00	G_4	392.00	G_5	783.99
A_2^b/G_2^{\sharp}	103.83	A_3^b/G_3^{\sharp}	207.65	A_4^b/G_4^{\sharp}	415.30	A_5^b/G_5^{\sharp}	830.61
A_2	110.00	A_3	220.00	A_4	440.00	A_5	880.00
B_2^b/A_2^{\sharp}	116.54	B_3^b/A_3^{\sharp}	233.08	B_4^b/A_4^{\sharp}	466.16	B_5^b/A_5^{\sharp}	923.33
B_2	123.47	B_3	246.94	B_4	493.88	B_5	987.77

Fase 2

Pada fase ini hasil pengenalan nada (*Chord Recognition*) pada *file* yang dimasukan tersebut diproses dan dikelompokkan berdasarkan rentangan jenis suara. Setelah dikelompokkan maka akan didapatkan hasil berupa informasi jenis suara dari pengguna. Misalkan dalam sebuah lagu seorang penyanyi lebih banyak menggunakan nada C₃, A₃, D₄ dan G₄ maka bisa disimpulkan penyanyi tersebut memiliki jenis suara bariton. Informasi jenis suara pengguna digunakan untuk menampilkan rekomendasi lagu. Lagu yang dipilih atau yang akan direkomendasikan adalah lagu yang penyanyinya memiliki jenis suara yang sama dengan pengguna. Gambar 2 adalah diagram kasus penggunaan dari sistem ini.



Gambar 2. Diagram Kasus Penggunaan

Pada kasus penggunaan memilih lagu, pengguna dapat memilih semua lagu yang disediakan sesuai dengan keinginan, selanjutnya pada kasus penggunaan mengoperasikan lagu, pengguna dapat melakukan *play*, *stop*, *next* dan *back* lagu. Pada kasus penggunaan memilih lagu berdasarkan rekomendasi, pengguna dapat memilih lagu yang ada pada daftar yang berisi rekomendasi lagu yang sesuai dengan jenis suara pengguna.

Contoh dari *Chord Recognition*

Contoh yang digunakan pada kasus *Chord recognition* ini adalah pengenalan dari tiga nada: (C₄, E₄, G₄), (C₄, E₄, G₄, A₄^h), (C₄, E₄, G₄, A₄, D₄) yang dimainkan pada piano, organ dan *flute*. Parameter rekaman yang digunakan adalah 16-bit *sample* yang diambil 44,100 *samples/s*. Panjang dari *input* adalah 2048 *sample*. Tabel 2, Tabel 3 dan Tabel 4 menunjukkan hasil *Chord Recognition* dari alat musik yang berbeda.

Tabel 2. Contoh dari *Chord Recognition* (Piano)

dB	C ₄ , E ₄ , G ₄	C ₄ , E ₄ , G ₄ , A ₄ [♯]	C ₄ , E ₄ , G ₄ , A ₄ , D ₅
0	G ₄	G ₄	D ₅
-1	G ₄	G ₄	G ₄
-2	G ₄	G ₄	G ₄
-5	C ₄	G ₄ , A ₄	G ₄ , D ₅
-6	C ₄ , C ₅	C ₄ , C ₅	G ₄ , D ₅
-7	C ₄ , C ₅	E ₄ , A ₄ , E ₅	G ₄ , D ₅
-8	E ₄ , F ₄ , E ₅	C ₄ , F ₄ , C ₅	E ₄ , G ₄ , D ₅
-9	<i>correct</i>	C ₄ , C ₄ [♯] , C ₅	F ₄ [♯] , G ₄ , D ₅ , G ₅
-10	<i>correct</i>	C ₄ , F ₄ , G ₄ [♯]	C ₄ , F ₄ [♯] , A ₄ , G ₄ , D ₅
-12	<i>correct</i>	C ₄ , F ₄ , G ₄ [♯]	E ₄ , G ₄ , C ₅
-13	<i>correct</i>	<i>correct</i>	E ₄ , G ₄ , C ₅
-14	C ₄ , C ₄ [♯] , C ₅	C ₄ , E ₄ , F ₄ [♯] , A ₄ [♯]	E ₄ , G ₄ , C ₅

Tabel 3. Contoh dari *Chord Recognition* (Organ)

dB	C ₄ , E ₄ , G ₄	C ₄ , E ₄ , G ₄ , A ₄ [♯]	C ₄ , E ₄ , G ₄ , A ₄ , D ₅
0	G ₄	G ₄	A ₅
-1	C ₄	C ₄	E ₄ , G ₄ , D ₅
-2	E ₄ , E ₅	E ₄ , E ₅	<i>correct</i>
-3	E ₄ , F ₄ , E ₅	C ₄ , C ₅	C ₄ , E ₄ , D ₅
-4	E ₄ , F ₄ , E ₅	C ₄ , C ₅	C ₄ , A ₄ , D ₅
-5	C ₄ , C ₄ [♯] , C ₅	C ₄ [♯] , G ₄ , A ₄	C ₄ , A ₄ , C ₅ , D ₅
-6	<i>Correct</i>	C ₄ , E ₄ , G ₄	C ₄ , E ₄ , G ₄ , A ₄
-8	<i>Correct</i>	<i>correct</i>	C ₄ , E ₄ , G ₄ , A ₄
-9	<i>Correct</i>	<i>correct</i>	<i>correct</i>
-10	<i>Correct</i>	<i>correct</i>	C ₄ [♯] , E ₄ , C ₅
-12	A ₃ [♯] , C ₄ [♯] , F ₄ [♯]	A ₃ [♯] , D ₄ , F ₄ , G ₄ [♯]	C ₄ [♯] , E ₄ , C ₅
-14	A ₃ [♯] , D ₄ , F ₄	C ₄ , E ₄ , F ₄ [♯] , A ₄ [♯]	E ₄ , C ₄ [♯] , A ₄ , C ₅ , D ₅

Tabel 4. Contoh dari *Chord Recognition* (Flute)

dB	C ₄ , E ₄ , G ₄	C ₄ , E ₄ , G ₄ , A ₄ [♯]	C ₄ , E ₄ , G ₄ , A ₄ , D ₅
0	G ₄	G ₄	D ₆
-1	G ₄	A ₄ [♯]	D ₆
-3	G ₄	G ₄	D ₆
-4	G ₄	G ₄ , G ₅	A ₅
-5	G ₄	G ₄ , G ₅	C ₄

-6	E ₄ , E ₅	G ₄ , A ₄ , G ₅	E ₄ , E ₅
-7	E ₄ , F ₄ , E ₅	F ₄ , G ₄ , A ₄ ^b , E ₅	E ₄ , E ₅
-8	E ₄ , F ₄ , E ₅	E ₄ , G ₄	E ₄ , E ₅
-10	E ₄ , G ₄	E ₄ , G ₄	C ₄ , C ₄ ^b , C ₅
-11	E ₄ , G ₄	E ₄ , F ₄ , B ₄	C ₄ , A ₄ , D ₅
-12	E ₄ , G ₄	E ₄ , F ₄ , G ₄ ^b , E ₅	C ₄ , A ₄ , C ₅ , D ₅
-13	E ₄ , G ₄	E ₄ , F ₄ , G ₄ ^b , E ₅	C ₄ , A ₄ , C ₅ , D ₅

Dari hasil proses *Chord Recognition* dapat disimpulkan bahwa tingkat ketepatan dalam mengenali nada dipengaruhi dari alat musiknya [2], begitu juga dalam mendeteksi nada pada suara penyanyi tentunya memiliki tingkat ketepatan yang berbeda.

10.METODOLOGI

a. Penyusunan Proposal Tugas Akhir

Pada tahap ini akan dilakukan penyusunan proposal Tugas Akhir yang merupakan langkah awal dalam mengerjakan Tugas Akhir ini. Proposal yang dibuat mengusulkan tentang aplikasi karaokae yang memiliki kelebihan fitur dapat memberikan rekomendasi lagu berdasarkan jenis suara.

b. Analisis Kebutuhan dan Studi Literatur

Pada tahap ini diperlukan adanya pengumpulan data dan pengumpulan informasi dari beberapa orang dan literatur yang diperlukan dalam proses perancangan dan implementasi sistem yang akan dibangun. Literatur yang digunakan adalah terkait dengan penerapan metode *Chord Recognition*.

c. Perancangan Sistem

Pada tahap ini akan dilakukan analisa awal dan dilakukan pendefinisian dari kebutuhan sistem yang dibangun untuk mengetahui masalah yang sedang dihadapi. Dari proses tersebut maka akan dilakukan perumusan rancangan sistem yang dapat memberikan solusi dari masalah yang dihadapi.

d. Implementasi

Pada tahap ini dilakukan pembuatan perangkat lunak yang merupakan implementasi dari rancangan yang telah dibuat sebelumnya. Rencana pembuatan perangkat lunak ini akan diimplementasikan dengan menggunakan:

1. Bahasa pemrograman yang digunakan adalah C#.
2. *Framework* yang digunakan adalah *framework* .NET 4.0.
3. *Tools* pemrograman yang digunakan adalah Visual Studio 2010.
4. *Tools* pemodelan yang digunakan adalah Star UML dan Power Designer.

e. Uji coba dan Evaluasi

Pada tahap ini akan dilakukan pengujian terhadap perangkat lunak yang dibangun dengan menggunakan data yang telah dipersiapkan sebelumnya. Uji coba dan evaluasi perangkat lunak ini digunakan untuk mencari kesalahan program yang timbul, mengevaluasi jalannya program, dan melakukan perbaikan apabila terjadi adanya kesalahan program.

f. Penyusunan Buku Tugas Akhir

Sistematika penulisan buku Tugas Akhir secara garis besar antara lain.

1. Pendahuluan
 - 1.1 Latar Belakang
 - 1.2 Permasalahan
 - 1.3 Batasan Tugas Akhir
 - 1.4 Tujuan
 - 1.5 Metodologi
 - 1.6 Sistematika Penulisan
2. Tinjauan Pustaka
3. Desain dan Implementasi
4. Uji Coba dan Evaluasi
5. Kesimpulan dan Saran
6. Daftar Pustaka

11. JADWAL KEGIATAN

Jadwal pengerjaan Tugas Akhir dapat dilihat pada Tabel 5.

Tabel 5. Rencana Jadwal Kegiatan

Tahapan	2014															
	Februari				Maret				April				Mei			
Penyusunan Proposal Tugas Akhir																
Studi Literatur																
Analisis dan Desain Perangkat Lunak																
Implementasi Perangkat Lunak																
Pengujian dan Evaluasi																
Penyusunan Buku Tugas Akhir																

12. DAFTAR PUSTAKA

- [1] T. Ken, "Ken Tamplin Vocal Academy," [Online]. Available:
<http://kentamplinvocalacademy.com/determining-singing-range-and-voice-type/>.
[Accessed 23 February 2014].
- [2] S. Marcin and . H. Wladyslaw, "A Practical Approach to the Chord Analysis in the Acoustical Recognition Process," in *Computational Intelligence for Modelling and Prediction*, vol. II, Berlin, Springer Berlin Heidelberg, 2005, p. 221–231.