

***FINAL REPORT***  
**PENELITIAN KOMPETENSI**



**"Pengembangan Metode Inferensi untuk  
Sistim Pakar berbasis *Ternary Grid*"**

**Dr. Ing. Yuliadi Erdani, M.Sc.**  
**Tahun Angkatan 2010**  
**Tahun ke 3**

**POLITEKNIK MANUFAKTUR NEGERI BANDUNG**  
**DESEMBER 2012**

LAMPIRAN 1: LEMBAR PENGESAHAN

PENELITIAN KOMPETENSI

1. Judul Kegiatan : “Pengembangan Metode Inferensi untuk Sistem Pakar berbasis Ternary Grid”  
2. Kata Kunci : Metode inferensi, sistem pakar, ternary grid  
3. Jenis Kegiatan : Penelitian Kompetensi  
4. Nama Ketua Tim Pengusul : Dr. Ing., Yuliadi Erdani, M.Sc.  
5. Jurusan : Teknik Otomasi Manufaktur dan Mekatronika  
Perguruan Tinggi : Politeknik Manufaktur Negeri Bandung  
6. Alamat : Jl. Kanayakan 21 Dago  
Bandung - 40135  
Telefon/ Faks : 022-250 0241/ 022- 2649  
E-mail : [yul\\_erdani@yahoo.com](mailto:yul_erdani@yahoo.com)  
7. Lamanya Kegiatan : 3 Tahun  
8. Nama dan Alamat Lengkap peers  
- dari dalam negeri : Prof. Dr. Harry Sudiby, Universitas Indonesia  
- dari luar negeri : Prof. Dr. Ing., Axel Hunger  
University of Duisburg-Essen, Germany  
9. Biaya yang disetujui  
Tahun III (tahun 2012) : Rp. 85.000.000,-

Mengetahui,  
Ketua Lembaga Penelitian/Pengabdian  
Kepada Masyarakat



(Dr. Ing., Yuliadi Erdani, M.Sc.)  
NIP. 1968 07 02 1997 02 1001

Bandung, 7 Desember 2012  
Ketua Tim Pelaksana,



(Dr. Ing., Yuliadi Erdani, M.Sc.)  
NIP. 1968 07 02 1997 02 1001

Mengetahui,  
Direktur Polman Bandung



(Prof. Dr. Ir. Isa Setiasyah Toha, M.Sc.)  
NIP. 130 808 003

DAFTAR ISI

LAMPIRAN 1: LEMBAR PENGESAHAN .....	1
DAFTAR ISI .....	2
RINGKASAN .....	3
1 PENDAHULUAN .....	4
2 KEGIATAN YANG SUDAH DILAKSANAKAN .....	4
3 KEBARUAN .....	5
4 RINGKASAN HASIL TAHUN KE 1 DAN 2 .....	5
4.1 Konsep Ternary Grid .....	5
4.2 Algoritma Inferensi .....	13
4.3 Database Facts & Rules .....	14
4.4 Inferensi dengan Forward Chaining .....	14
5 METODOLOGI PELAKSANAAN .....	21
5.1 <i>Backward Chaining</i> .....	21
5.2 Metode .....	22
5.3 Roadmap .....	29
6 HASIL KEGIATAN TAHUN KE 3 .....	32
7 KESIMPULAN .....	39
8 ORGANISASI TIM PENELITI .....	40
9 JADWAL KEGIATAN .....	40
10 PENDANAAN .....	41
10.1 Pendanaan Tahun ke 3 (2012) .....	41
10.2 Justifikasi Pendanaan .....	42
DAFTAR PUSTAKA .....	43
LAMPIRAN 2: SUSUNAN ORGANISASI .....	47
LAMPIRAN 3: KETERSEDIAAN SARANA DAN PRASARANA PENELITIAN .....	48
LAMPIRAN 4: BIODATA KETUA .....	49
LAMPIRAN 5: PUBLIKASI .....	56
LAMPIRAN 6: LAPORAN KEUANGAN (BUKU KAS UMUM) .....	62

**Judul:****“Pengembangan Metode Inferensi untuk Sistem Pakar berbasis Ternary Grid”****RINGKASAN**

Sistem Pakar berbasis aturan merupakan sistem pakar yang berkembang sangat pesat dan banyak diimplementasikan pada berbagai keperluan baik di bidang teknik maupun non teknik. Pada kebanyakan Sistem Pakar berbasis Aturan (rule-based expert system), pembuatan aturan (Rule) dapat dilakukan dengan cepat tanpa harus memikirkan spesifikasi aturan dan bagaimana aturan tersebut terhubung satu sama lainnya. Seringkali seorang expert mereferensikan pengetahuannya kepada aturan atau fakta yang sama sekali belum dibuat. Pekerjaan tersebut tampaknya mudah dan sederhana. Permasalahan akan timbul apabila jumlah aturan menjadi banyak. Beberapa masalah dapat timbul dalam bentuk aturan yang tidak konsisten, aturan yang tidak pernah bisa diraih, aturan redundan, aturan melingkar, dll.

Model pengetahuan Ternary Grid merupakan salah satu model pengetahuan baru yang dapat menawarkan penyelesaian masalah tersebut. Hingga saat ini belum ada metode inferensi untuk sistem pakar yang bekerja pada model Ternary Grid tersebut. Pada penelitian ini diusulkan aktifitas ilmiah untuk menemukan Teknik Inferensi berbasis Ternary Grid yang berimplikasi kepada pengembangan sistem pakar berbasis Ternary Grid. Implementasi dari keseluruhan sistem tersebut berupa perangkat lunak.

Pada tahun pertama, tim peneliti telah memperbaiki dan menyempurnakan metode inferensi yang pernah dikembangkan sebelumnya, yaitu metode inferensi dengan pendekatan interaktif pada Ternary Grid. Sebagai pendukungnya, tim peneliti telah membuat database fakta dan aturan (facts & rules) berbasis Ternary Grid. Untuk meningkatkan efisiensi proses inferensi, Berdasarkan hasil yang didapat pada tahun pertama, pada tahun kedua ini tim peneliti telah mengembangkan metode inferensi dengan pendekatan rekursif dan mengimplementasikannya kedalam bentuk simulator inferensi. Untuk mengintegrasikan seluruh software yang dikembangkan dan mensinkronkan metode-metode inferensi yang dikembangkan, pada tahun ketiga tim peneliti akan mengembangkan solusi lainnya dengan backward chaining dan pendekatan rekursif.

Berdasarkan hasil rancangan yang telah dikembangkan, proses konversi dari pengetahuan berbasis aturan ke pengetahuan ternary grid tidak diperlukan lagi sehingga eksekusi sistem pakar menjadi efisien. Demikianpun dengan duplikasi aturan pada basis pengetahuan dapat dihindari karena pada ternary grid dimungkinkan pendeteksian duplikasi aturan. Strategi yang digunakan untuk mencari solusi adalah forward chaining dengan pendekatan iteratif. Implementasi dari keseluruhan sistem tersebut berupa perangkat lunak. Hasil eksperimen menunjukkan bahwa proses inferensi berjalan sempurna dan lebih efisien dibandingkan dengan proses inferensi pada sistem pakar berbasis aturan yang telah dikembangkan sebelumnya, baik dalam proses inferensi maupun kinerja sistem pakar.

Merujuk kepada beberapa literatur yang berkaitan dengan sistem pakar atau sistem basis pengetahuan, metode inferensi dengan Ternary Grid ini merupakan hal baru dalam pengembangan teknik inferensi sehingga diharapkan dapat memberikan kontribusi yang signifikan bagi pengembangan sistem pakar atau sistem basis pengetahuan.

Kata kunci: metode inferensi, backward chaining, sistem pakar, ternary grid

## **1 PENDAHULUAN**

Sistim pakar merupakan bagian dari sistim cerdas yang aplikasinya dapat diterapkan di berbagai bidang mulai dari teknik, ekonomi, sosial, pendidikan [Yuliadi, 2007a], dll. Salah satu aplikasi sistim Basis Pengetahuan yang pernah dibuat adalah CongaXpert [Erdani, 2003a] [Erdani, 2003b]. CongaXpert merupakan software aplikasi berbasis web yang bertugas sebagai konsultan pendidikan bagi para mahasiswa dan calon mahasiswa Universitas Duisburg-Essen, Jerman.

Kelemahan dari sistim tersebut terletak pada Basis Pengetahuan yang masih menggunakan format orisinal sistim produksi yang dikembangkan oleh [Newell, 1972]. Meskipun format aturan tersebut masih merupakan format aturan baku untuk Sistim Pakar berbasis Aturan, namun format tersebut menimbulkan kesulitan pada saat proses akuisisi pengetahuan. Akibat dari kelemahan ini yaitu turunnya kualitas pengetahuan hasil akuisisi.

Sebagai salah satu solusi untuk menyelesaikan masalah tersebut telah diperkenalkan Sistim Pengakuisisian Pengetahuan berbasis Ternary Grid [Erdani, 2004]. Baik metoda akuisisi maupun Model Pengetahuan dengan Ternary Grid terbukti dapat memperbaiki kinerja pengetahuan yang implikasinya berupa perbaikan kualitas pengetahuan yang dihasilkan dari proses akuisisi [Erdani 2005a] [Erdani, 2005b] [Erdani, 2005c].

Meskipun sudah ada perbaikan dalam metoda akuisisi dan model pengetahuan, namun ditinjau dari sisi Sistim Pakar masih ada kekurangan yaitu pada bagian Teknik Inferensi. Hingga saat ini belum ada Teknik Inferensi Sistim Pakar yang bekerja pada model tersebut sehingga Model Pengetahuan berbasis Ternary Grid masih harus dikonversi balik ke format sistim produksi [Newell, 1972] yang selanjutnya diproses oleh mesin inferensi untuk menghasilkan kesimpulan.

## **2 KEGIATAN YANG SUDAH DILAKSANAKAN**

Berkaitan dengan kegiatan yang diusulkan pada proposal ini, peneliti sudah melakukan berbagai kegiatan dari tahun 2003 hingga 2009. Pada tahun 2003, peneliti melakukan pengembangan sistem pakar berbasis web untuk keperluan konsultasi akademik dan mempublikasikannya pada seminar internasional [Erdani,2003a] dan [Erdani, 2003b]. Selanjutnya penelitian melakukan pengembangan pada komponen lainnya dari sistem pakar, yaitu sistem akuisisi dan elisitasi, dan mempublikasikan pada seminar internasional yang cukup kompeten [Erdani, 2004]. Pada tahun 2005, sistem akuisisi sistem pakar yang dikembangkan peneliti mendapat pengakuan internasional dari WSEAS Transaction [Erdani, 2005b] dan mempublikasikannya pada seminar yang dilakukan oleh WSEAS [Erdani, 2005a]. Pada [Erdani, 2005c], peneliti diminta untuk mendiseminasikan metode akuisisi sistem pakar yang sudah dikembangkannya. Pada tahun itu pula, peneliti menuliskan seluruh pemikiran berkaitan dengan pengembangan sistem akuisisi sistem pakar dalam bentuk buku [Erdani, 2005d].

Sepulang studi di luar negeri, peneliti terus tidak berhenti melanjutkan penelitian dan pengembangan di bidang sistem pakar. Mulai dari tahun 2007 sampai dengan tahun 2009, peneliti mendapatkan dana bantuan dari Pemerintah Republik Indonesia

melalui skim Hibah Bersaing DP2M-DIKTI untuk melakukan penelitian dan pengembangan di bidang sistem pakar. Pada tahun 2007, peneliti mempublikasikan karya penelitiannya pada berbagai seminar nasional [Yuliadi, 2007a], [Yuliadi 2007b], [Yuliadi 2007c]. Hasil penelitian yang didanai oleh Hibah Bersaing dapat diketahui pada akhir tahun pertama pendanaan dan dipublikasikan pada tahun 2008 [Yuliadi, 2008a]. Hasil penelitian tersebut yaitu berupa database basis pengetahuan berbasis Ternary Grid. Pada tahun yang sama, peneliti memulai pengembangan teknik inferensi, meskipun hasilnya masih berupa konsep karena keterbatasan kemampuan, dana dan waktu. Hasil tersebut dapat dipublikasikan pada [Yuliadi, 2008b]. Pada tahun 2009, peneliti mengembangkan konsep shell sistem pakar berbasis ternary Grid [Yuliadi, 2009] yang merupakan salah satu komponen penting pada sistem pakar. Pada 2010, peneliti berhasil mengembangkan metode inferensi pada Ternary Grid dengan pendekatan iteratif dan telah mempublikasikan hasil karyanya di jurnal nasional [Yuliadi, 2010]. Pada tahun 2011 peneliti berhasil mengembangkan metode inferensi dengan metode forward chaining melalui pendekatan rekursif yang diterapkan pada ternary grid [Yuliadi, 2011].

Sebagai kelanjutan dari kegiatan-kegiatan penelitian yang berkaitan dengan Ternary Grid, pada penelitian yang diusulkan ini peneliti bermaksud melakukan aktifitas ilmiah dalam rangka mengembangkan dan menyempurnakan metode inferensi sistem pakar berbasis Ternary Grid yang diimplementasikan dalam bentuk perangkat lunak.

### **3 KEBARUAN**

Mesin inferensi merupakan salah satu komponen utama yang membentuk sistem pakar. Sejak ide Ternary Grid digulirkan [Erdani, 2004] hingga saat ini, belum ditemukan atau dikembangkan metode, teknik atau mesin inferensi sistem pakar yang bekerja pada model Ternary Grid tersebut. Sebagai konsekuensinya, model pengetahuan berbasis Ternary Grid masih harus dikonversi balik ke format sistem produksi [Newell, 1972] agar pengetahuan tersebut dapat diproses oleh mesin inferensi untuk menghasilkan kesimpulan (output) sistem pakar.

Melalui kegiatan penelitian ini, tim peneliti bermaksud mengembangkan metode inferensi sistem pakar yang bekerja pada model pengetahuan Ternary Grid. Dengan harapan yang besar, peneliti berusaha dan bekerja keras untuk mengembangkan atau menemukan metode inferensi tersebut. Dengan ditemukannya metode inferensi yang dimaksud, maka metode ini merupakan metode yang baru (novel) yang memperkaya metode-metode yang selama ini ada dan bekerja pada model pengetahuan yang sudah ada pula. Tentunya pula, hal ini merupakan suatu kontribusi yang sangat berarti (signifikan) bagi pengembangan sistem pakar atau sistem cerdas lainnya yang menggunakan metode inferensi di dalam proses reasoning-nya.

### **4 RINGKASAN HASIL TAHUN KE 1 DAN 2**

#### **4.1 Konsep Ternary Grid**

Sebagai bagian dari Sistem Basis Pengetahuan atau Sistem Pakar, Mesin Inferensi (inference engine) bertugas melakukan inferensi atau *reasoning* terhadap aturan-aturan yang disimpan pada Basis Pengetahuan. Mesin inferensi merupakan otaknya

sistim basis pengetahuan yang mengolah informasi dari basis pengetahuan. Cara kerja dari mesin inferensi adalah mengolahnya fakta yang diberikan oleh pengguna dan mencari keterkaitan antara fakta-fakta tersebut dengan fakta-fakta dan aturan-aturan yang disimpan pada basis pengetahuan. Terdapat dua teknik yang digunakan dalam melakukan inferensi, yaitu *forward chaining* dan *backward chaining* [Lunze, 1994].

Pengetahuan yang disimpan di dalam Basis Pengetahuan dapat diilustrasikan seperti terlihat pada gambar 10. Basis pengetahuan yang digunakan oleh Sistem Pakar pada penelitian ini memiliki format aturan produksi [Newell, 1972] yang komposisinya terdiri dari Fakta dan Aturan. Format tersebut telah dioptimalkan pada penelitian terdahulu [Erdani, 2005a] [Erdani, 2005b] [Erdani, 2005c] [Yuliadi, 2007c] agar didapatkan kinerja pengetahuan yang lebih baik. Format tersebut selanjutnya disebut sebagai Basis Pengetahuan Ternary Grid.

Ternary Grid merupakan susunan pengetahuan yang dikembangkan dalam format grid atau matriks. Baris dari matriks merepresentasikan Aturan, sementara kolom dari matriks merepresentasikan Fakta. Nilai setiap sel matriks yang memungkinkan hanya 0, 1 dan 2. Nilai 1 mengindikasikan bahwa suatu Fakta yang terkait merupakan bagian kondisi dari Aturan yang terkait. Nilai 2 mengindikasikan bahwa Fakta yang terkait merupakan bagian simpulan dari Aturan yang terkait. Sementara nilai 0 mengindikasikan antara Fakta dan Aturan tidak ada keterkaitan (Fakta tersebut bukan bagian dari Aturan). Gambar 1 menunjukkan matriks dari Ternary Grid.

	<i>F1</i>	<i>F2</i>	...	<i>FJ</i>
<i>R1</i>	$a_{11}$	$a_{12}$		$a_{1j}$
...				
<i>RI</i>	$a_{i1}$	$a_{i2}$		$a_{ij}$

Gambar 1. Notasi matriks Ternary Grid

Dimana:

$$i = \{1, \dots, I, I \in \mathbb{N}\}$$

$$j = \{1, \dots, J, J \in \mathbb{N}\}$$

Sementara  $J \geq I + 1$  dan  $a_{ij} = \{0, 1, 2\}$

Nilai 0 direpresentasikan sebagai sel matriks yang kosong.

Dengan struktur matriks tersebut, maka beberapa himpunan yang ada pada matriks dapat dijelaskan dengan rumus matematika sebagai berikut:

- Himpunan bagian kondisi untuk setiap baris dirumuskan sebagai berikut:

$$Ri1 = \{ j \mid a_{ij} = 1 \}$$

- Himpunan bagian simpulan untuk setiap baris dirumuskan sebagai berikut:

$$Ri2 = \{ j \mid a_{ij} = 2 \}$$

- Himpunan bagian kondisi untuk setiap kolom dirumuskan sebagai berikut:

$$Fj1 = \{ i \mid a_{ij} = 1 \}$$

- Himpunan bagian simpulan untuk setiap kolom dirumuskan sebagai berikut:

$$Fj2 = \{ i \mid a_{ij} = 2 \}$$

Bagi para pakar pengetahuan, rumus-rumus di atas tersebut dapat dinyatakan sebagai berikut:

- Himpunan Aturan

$$R = \{ R1, \dots, RI \}$$

- Himpunan fakta-fakta yang merupakan bagian kondisi dari Aturan Ri

$$Ri1 = \{ Fj \mid a_{ij} = 1 \}$$

- Himpunan fakta-fakta yang merupakan bagian simpulan dari Aturan Ri

$$Ri2 = \{ Fj \mid a_{ij} = 2 \}$$

- Himpunan Fakta

$$F = \{ F1, \dots, FJ \}$$

- Himpunan aturan-aturan yang memuat fakta-fakta bagian kondisi Fj

$$Fj1 = \{ Ri \mid a_{ij} = 1 \}$$

- Himpunan aturan-aturan yang memuat fakta-fakta bagian simpulan Fj

$$Fj2 = \{ Ri \mid a_{ij} = 2 \}$$

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14
R1	1	1	2	0	0	0	0	0	0	0	0	0	0	0
R2	0	0	1	2	0	0	0	0	0	0	0	0	0	0
R3	0	0	0	1	1	2	0	0	0	0	0	0	0	0
R4	0	2	0	0	0	0	1	1	0	0	0	0	0	0
R5	0	2	0	0	0	0	1	1	1	0	0	0	0	0
R6	0	3	0	0	0	0	0	0	0	1	1	0	0	0
R7	0	2	0	0	0	0	0	0	0	0	0	1	1	0
R8	1	1	0	0	0	0	0	0	0	0	0	0	0	2
R9	0	0	0	1	1	2	0	0	0	0	0	0	0	0

Gambar 2. Contoh kasus, aturan dalam format Ternary Grid

Sebelum masuk ke tahap inferensi, semua aturan akan dioptimalkan dan dikonversi ke format Ternary Grid [Erdani, 2005a] [Erdani, 2005b] [Erdani, 2005c] [Yuliadi, 2007c]. Agar proses inferensi dapat dipahami dengan baik, maka pada bagian ini diulas lagi dengan ringkas (*brief review*) proses akuisisi dari suatu aturan. Untuk lebih jelasnya, diberikan kasus yang berisi aturan dalam format Ternary Grid seperti terlihat pada gambar 2. Dengan merepresentasikan nilai 0 sebagai sel grid atau kotak matriks yang kosong, maka susunan matriks aturan terlihat pada gambar 3.

	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14
R1	1	1	2											
R2			1	2										
R3				1	1	2								
R4		2					1	1						
R5		2					1	1	1					
R6		3								1	1			
R7		2										1	1	
R8	1	1												2
R9				1	1	2								

Gambar 3. Eliminasi nilai 0 pada matriks aturan

Selanjutnya aturan-aturan yang ada pada matriks tersebut dioptimalkan dengan algoritma optimasi aturan Ternary Grid. Optimalisasi meliputi peniadaan beberapa aturan dengan memperhatikan kepada beberapa sifat atau karakter aturan yang meliputi:

- Konsistensi dan kelaikan (*consistency and feasibility*)

Aturan yang tidak konsisten adalah aturan yang tidak memiliki makna jelas. Hal ini terjadi apabila bagian simpulan suatu aturan dijadikan lagi sebagai bagian kondisi aturan.

Aturan yang tidak laik adalah aturan yang tidak akan pernah bisa diaplikasikan, sebab bagian simpulan dari aturan tidak akan pernah terpenuhi dikarenakan bagian kondisinya juga tidak akan pernah menghasilkan nilai benar (terpenuhi).

Baik aturan yang tidak konsisten maupun aturan yang tidak laik memiliki struktur aturan sebagai berikut:

$$IF \langle \dots \rangle \& \langle \dots \rangle \& \langle Y \rangle THEN \langle Y \rangle + \langle \dots \rangle + \langle \dots \rangle$$

Aturan tersebut dapat diinvestigasi dengan algoritma sebagai berikut:

- o Mencari aturan-aturan yang memiliki nilai 3

$$B = \{ i \mid a_{ij} = 3 \} \text{ atau } B = \left\{ i \mid \left( \sum_j (a_{ij} \text{ div } 3) \right) > 0 \right\}$$

Aturan tersebut tidak dapat langsung dieliminasi, melainkan diserahkan kepada pengguna untuk memperbaikinya, karena maksud atau konten suatu aturan hanya pengguna yang tahu. Mesin hanya bisa memberikan informasi bahwa susunan aturan tidak valid.

- Redundansi (redundancy)

Pembahasan yang berkaitan dengan redundansi adalah investigasi dan eliminasi terhadap aturan-aturan yang memiliki beberapa sifat sebagai berikut:

- o Aturan yang memiliki fakta berulang (*repeating fact*)

Pada aturan ini, fakta yang identik muncul secara berulang (lebih dari satu kali) di bagian kondisinya. Jika tidak diatasi, maka nilai grid atau matriks pada Ternary Grid akan terakumulasi sehingga nilai akhirnya tidak valid. Untuk mengatasinya, pada saat penulisan nilai grid atau matriks, nilai baru dioperasikan dengan nilai aktual (*existing value*) menggunakan operator logika OR, sehingga nilai yang baru akan senantiasa valid. Secara matematik operasi tersebut dapat dituliskan sebagai berikut:

$$y_n = \max(y_{n-1}, x) = y_{n-1} + x = y_{n-1} \text{ OR } x$$

$y_n$  : nilai akhir

$y_{n+1}$  : nilai aktual

$x$  : nilai baru

- o Aturan berulang (*repeating rule*)

Aturan berulang adalah aturan yang identik muncul kembali (digenerasi) oleh pengguna. Aturan berulang tersebut dapat dihilangkan dengan algoritma berikut:

- Mencari fakta-fakta dimana nilai 2 muncul:

$$B_j = \{b \mid b \in F_j^2, |F_j^2| \geq 2\}$$

- Membandingkan aturan-aturan yang bagian kondisinya identik:

$$C_j = \{(p, q) \mid p \in B_j, q \in B_j, p < q, |B_j| > 0, |R_{p1}| = |R_{q1}|\}$$

- Mengeliminasi aturan-aturan yang redundan

$$C = \bigcup_{j \in N} C_j$$

$$D = \{(p, q) \mid (p, q) \in C, R_{p1} = R_{q1}\}$$

- Aturan yang memiliki bagian kondisi redundan

Aturan ini memiliki bagian simpulan yang sama dengan aturan lain, namun bagian kondisinya sudah terwakili oleh aturan lain tersebut. Dengan demikian aturan tersebut dapat dieliminasi. Eliminasi aturan ini dapat dijelaskan dengan algoritma berikut:

- Mencari fakta-fakta dimana terdapat nilai 2

$$B_j = \{b \mid b \in F_j^2, |F_j^2| \geq 2\}$$

- Mencari baris-baris yang nilai 1-nya berada pada kolom yang berlainan

$$C_j = \{(p, q) \mid p \in B_j, q \in B_j, p < q, |B_j| > 0, |R_{p1}| \neq |R_{q1}|\}$$

$$C = \bigcup_{j \in N} C_j$$

- Mencari baris-baris dimana bagian kondisinya merupakan himpunan bagian dari bagian kondisi aturan lain

$$D = \{(p, q) \in C, R_{p1} \in R_{q1} \text{ or } R_{q1} \in R_{p1}\}$$

$$E = \{p \mid p \in (p, q), (p, q) \in D, R_{q1} \in R_{p1}\} \text{ atau}$$

$$E = \{ q \mid q \in (p, q), (p, q) \in D, Rp1 \in Rq1 \}$$

o Aturan transitif

Aturan transitif adalah aturan yang memenuhi prinsip transitif (*the principle of the equality*) dimana berlaku  $x = y$  dan  $y = z$ ; maka  $x = z$ . Dengan adanya aturan transitif maka perlu digenerasi aturan baru agar proses inferensi dapat lebih efisien.

- Mencari pasangan aturan dan fakta dimana nilai 1 muncul hanya satu kali

$$B = \{ (i, j) \mid a_{ij} = 1, |Ri1| = 1 \}$$

- Mencari aturan-aturan dimana bagian simpulannya merupakan bagian kondisi dari aturan lain.

$$C = \{ x \mid x \in Fj2, j \in (i, j), (i, j) \in B \}$$

- Generasi aturan baru

$$Rn1 = \{ j \mid j \in Ri1, n \in \mathbb{N}, i \in C \}$$

$$Rn2 = \{ j \mid j \in Ri2, n \in \mathbb{N}, i \in B \}$$

o Aturan yang memiliki bagian kondisi berulang

Aturan ini sulit dideskripsikan, namun dapat dijelaskan dengan notasi sebagai berikut:

R1: IF <f1> and <f2> THEN <f3>

R2: IF <f1> and <f2> and <f3> THEN <f4>

- Mencari aturan dimana bagian kondisi dan simpulannya merupakan himpunan bagian dari aturan lain

$$Bi = \{ p \mid |Ri1| \geq |Rp1| + |Rp2|, i \neq p, p \in \mathbb{N} \}$$

$$Ci = \{ p \mid (Rp1 \cup Rp2) \in Ri1, i \neq p, p \in Bi \}$$

- Mengenerasi aturan baru yang mewakili aturan-aturan tersebut di atas

$$Rn1 = \{ p \mid p \neq q, p \in Ri1, q \in Rs2, s \in Ci, n \in N \}$$

$$Rn2 = \left\{ p \mid p \in \left( Ri2 \cup \bigcup_{q \in Ci} Rq2 \right), n \in N \right\}$$

- Simpulan jamak (multiple conclusion)

Aturan ini memiliki bagian kondisi yang sama dengan aturan lainnya namun berbeda pada bagian simpulannya. Dengan demikian kedua aturan tersebut dapat dieliminasi dan diganti dengan aturan yang baru yang mewakili keduanya.

- Mencari aturan-aturan yang memiliki bagian kondisi yang sama

$$Bi = \{ p \mid Ri1 = Rp1, i < p, p \in N \}$$

- Membuat aturan baru  $Rq$  yang memiliki bagian simpulan jamak

$$Rq2 = \{ Ri2 \cup Rp2 \mid p \in Bi, q \in N \} \text{ dan } Rq1 = Ri1$$

- Eliminasi aturan-aturan lama

$$R = \{ Ri \cup Rp \mid p \in Bi \}$$

- Rantai berputar (endless loop/rotating chain)

Aturan yang memiliki sifat rantai berputar adalah aturan yang paling kritis, karena tidak akan menghasilkan keputusan yang pasti dan akan mengakibatkan mesin inferensi tidak bisa berhenti bekerja (*hang*). Aturan ini terjadi apabila bagian simpulan suatu aturan menjadi bagian kondisi dari aturan lainnya, dan simpulan aturan lain tersebut menjadi bagian kondisi dari aturan sebelumnya. Aturan ini tidak dapat dieliminasi, melainkan harus direkonstruksi ulang oleh pengguna, karena konten aturan hanya pengguna yang tahu.

- Mencari fakta-fakta dimana hanya nilai 2 muncul

$$Bj = \{ b \mid b \in Fj2, |Fj2| \geq 1, |Fj1| = 0 \}$$

Jika  $Bj = 0$  maka basis pengetahuan tersebut terdapat aturan yang memiliki sifat rantai berputar.

Setelah proses optimasi aturan tersebut dilakukan, maka dihasilkan aturan-aturan yang sudah dioptimalkan seperti terlihat pada gambar 4.

	<i>F1</i>	<i>F2</i>	<i>F3</i>	<i>F4</i>	<i>F5</i>	<i>F6</i>	<i>F7</i>	<i>F8</i>	<i>F9</i>	<i>F10</i>
<b>R1</b>	1	1	2							2
<b>R2</b>			1	1	2					
<b>R3</b>		2				1	1			
<b>R4</b>		2						1	1	

Gambar 4. Aturan-aturan yang sudah dioptimalkan

## 4.2 Algoritma Inferensi

Algoritma yang dikembangkan dengan mengacu kepada konsep tersebut dijelaskan sebagai berikut:

- *Make list of known facts*  $F_x = \{F_p, \dots, F_q\}$
- *Make list of possible applicable rules corresponding to known facts*  
 $R_x = \{R_m, \dots, R_n\}$
- *For*  $i = m$  *to*  $n$  *Do*
  - *For*  $j = m$  *to*  $n$  *Do*
    - *IF*  $R_i$  *applicable THEN*
      - *Add*  $R_i$  *to list*  $R_{x_i}$
      - *Add new known fact*  $F_y$  *corresponding to*  $R_i$  *to list*  
 $F_x$
  - $R_{x_i} = \{R_{x_i}, R_{x_{i-1}}\}$
  - *IF*  $|R_{x_i}| \geq n$  *THEN exit from loop*

Pertama-tama fakta-fakta yang dikenal dihimpun ke dalam suatu daftar (list). Dari himpunan fakta-fakta tersebut didapatkan himpunan aturan-aturan yang berkorespondensi dengan fakta-fakta tersebut yang selanjutnya dihimpun di dalam suatu daftar pula. Aturan-aturan tersebut merupakan aturan yang berpotensi untuk digunakan. Selanjutnya dilakukan proses iterasi ganda sebanyak jumlah aturan tersebut. Setiap aturan yang dapat digunakan, disimpan pada himpunan baru. Demikianpun dengan fakta baru yang dikenal disimpan di dalam himpunan baru. Proses iterasi akan berhenti apabila jumlah anggota himpunan aturan yang bisa digunakan sudah mencukupi. Hal ini mengindikasikan tidak ada aturan lagi yang memungkinkan untuk digunakan, dikarenakan sudah tidak ada fakta lagi yang berkorespondensi dengan aturan tersebut.

### **4.3 Database Facts & Rules**

Kualitas pengetahuan tergantung dari kualitas struktur database. Struktur database harus mampu menyimpan pengetahuan dalam bentuk data dan mengekstraksinya kembali jika data-data tersebut ditransforasikan ke bentuk pengetahuan lagi tanpa adanya informasi yang hilang.

Struktur database yang memenuhi syarat hanya bisa dicapai dengan perancangan database yang baik yang memperhatikan karakter-karakter tersebut. Perancangan database berisi perancangan struktur logika dan fisik untuk mengakomodasi informasi yang diperlukan oleh pengguna pada sekumpulan aplikasi. Perancangan database pada penelitian ini berisi tahapan sebagai berikut:

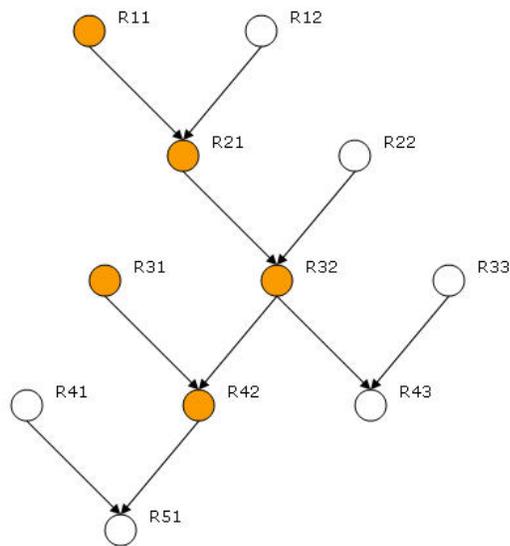
- Perencanaan dan analisa
- Perancangan konsep
- Perancangan logika
- Perancangan fisik
- Implementasi

Salah satu bagian terpenting dari proses perancangan database adalah pembuatan model data. Model data berisi informasi bagaimana suatu data disimpan di dalam database dan digunakan untuk merancang tabel relasi. Pemodelan data ini merupakan bagian yang membutuhkan waktu dan tenaga terbesar dalam seluruh proses pengembangan database. Tujuan dari pemodelan data adalah untuk memberikan jaminan terhadap keakurasian dan kelengkapan data yang direpresentasikan. Salah satu metoda yang terkenal dan banyak dipakai dalam pemodelan data adalah Entity-Relationship Model (ERM), yaitu model relasi antar entitas (tabel).

Sesuai dengan rancangan yang sudah dibuat, maka database yang dikembangkan berisi tiga sub domain, yaitu sub domain fakta, sub domain aturan dan sub domain lainnya.

### **4.4 Inferensi dengan Forward Chaining**

Proses forward chaining dengan rekursif dapat diilustrasikan seperti terlihat pada gambar 5 di bawah ini:



Gambar 5. Ilustrasi forward chaining

Fakta R11, R21, R31, R32 dan R42 adalah fakta-fakta yang sudah dikenal. Proses inferensi dimulai dari fakta R11 dan maju terus ke bagian kesimpulan. Dari gambar 24 dapat dilihat hanya satu aturan yang dapat digunakan, dimana R42 sebagai bagian kesimpulan dan R31 dan R32 sebagai bagian kondisi dari aturan. Dengan cara yang sama, proses inferensi dapat diterapkan pula pada basis pengetahuan berbasis Ternary Grid.

Algoritma proses inferensi dengan forward chaining pada basis pengetahuan Ternary Grid dijelaskan pada notasi algoritma berikut:

```

Input known facts
Mark Rootnode
Searchnode = Rootnode
Searchbegin = 1

Do while there is applied rule or the queue is not
empty
  Search step
  Start search from rule Searchbegin according to a
  rule number x, which is applicable in the
  Searchnode and leads to an unknown Nextnode
  If there is applied rule then
    Continue search
    Store Nextnode in dMark
    Searchbegin = 1
  Else if the queue is not empty
    Change Searchnode
    Searchnode = Endnode
    of the first edge of the queue
    Searchbegin = 1
    Advance the queue

Result output

```

Data-data untuk ujicoba pada proses inferensi ini sama dengan data-data yang digunakan pada [Yuliadi, 2010], seperti terlihat pada aturan di bawah ini:

```

IF <tutorial score >= 60> AND <thesis draft is
finished> THEN <tutorial result passed>; IF <
tutorial result passed> AND <presentation
score greater than or equals 60> AND
<comprehensive test passed> THEN <final
project passes>; IF <final project passes> AND
<thesis revision on time> THEN < graduation
on time>; IF < final project passes > AND
<thesis revision not on time > THEN
<graduation postponed>

```

Sesuai dengan teknik akuisisi Ternary Grid sebagaimana tercantum pada [Erdani, 2005], aturan-aturan tersebut dimasukkan ke dalam basis pengetahuan Ternary Grid seperti terlihat pada gambar 25. Dengan konsep yang dikembangkan, format berbasis aturan tidak perlu lagi dikonversi ke dalam format Ternary Grid. Selanjutnya dilakukan proses inferensi forward chaining secara rekursif. Semua fakta-fakta masukan disimpan di dalam himpunan fakta  $F_k$ . Kemudian mesin inferensi mencari seluruh aturan yang memungkinkan untuk digunakan atau dieksekusi, dan selanjutnya disimpan di dalam himpunan  $R_x$  sebagai berikut:

$$R_x = \{p \mid p \rightarrow q, p \in F_k, p \in F, F_k \subset F\}$$

Mesin inferensi menentukan aturan-aturan yang dapat digunakan dan selanjutnya disimpan di dalam himpunan  $R_{y_n}$  berikut:

$$R_{y_n} \subset R_x$$

Rule\Fact	F 1 (24)	F 2 (25)	F 3 (26)	F 4 (27)	F 5 (28)	F 6 (29)	F 7 (30)	F 8 (31)	F 9 (32)	F 10 (33)
R 1 (6)	1	1	2							
R 2 (7)			1	1	1	2				
R 3 (8)						1	1		2	
R 4 (9)						1		1		2

Gambar 6. Aturan dalam format Ternary Grid

Existing Facts: Existing Facts

List of User Facts: Clear List Save as User Fact in the List below

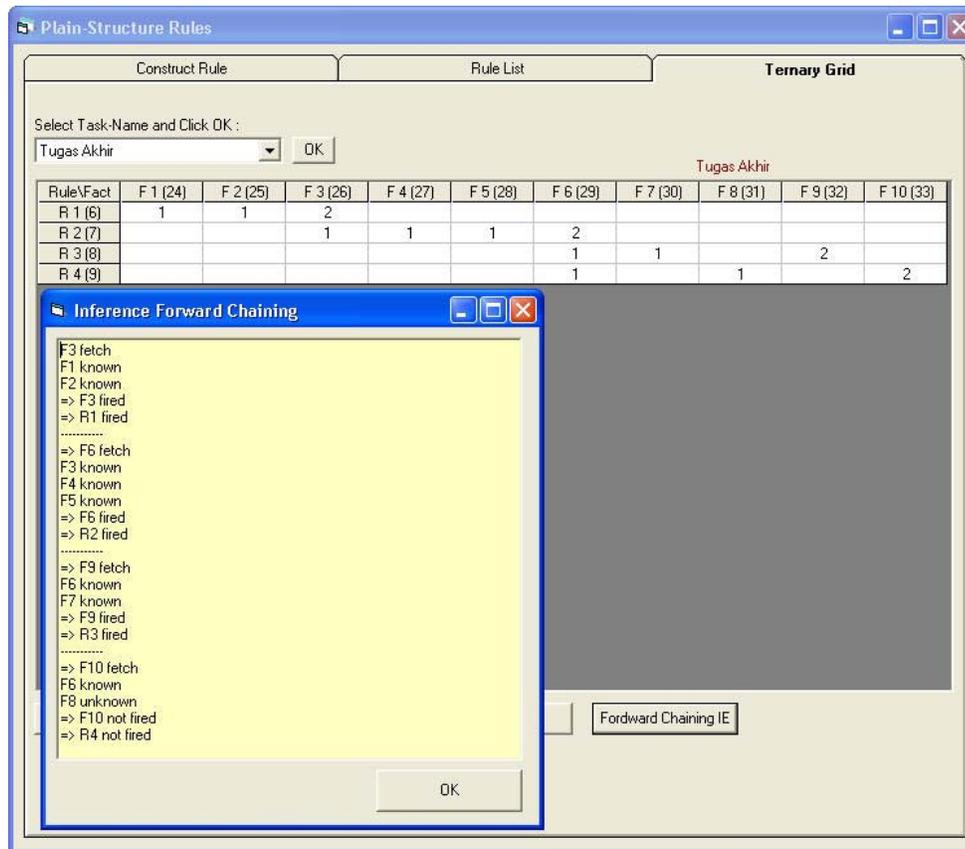
- (F1) Nilai Bimbingan >= 60
- (F2) Draft Karya Tulis Selesai
- (F4) Nilai Presentasi >= 60
- (F5) Test komprehensif Lulus
- (F7) Revisi Draft Karya Tulis Selesai Tepat Waktu

OK

Gambar 7. Fakta-fakta yang sudah dikenali

Untuk menghindari duplikasi, rumus berikut diimplementasikan di dalam mesin inferensi.

$$R_z = \bigcup_n R_{y_n}$$



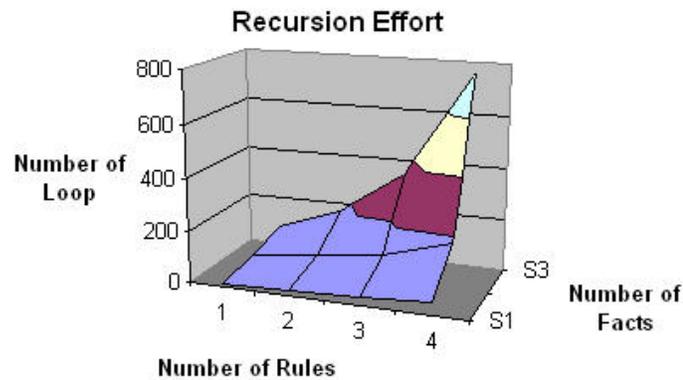
Gambar 8. Hasil proses inferensi dengan forward chaining

Hasil proses inferensi menunjukkan keefektifan algoritma yang dikembangkan di dalam penelitian ini. Dibandingkan dengan metode yang dikembangkan sebelumnya oleh [Yuliadi, 2008] [Yuliadi, 2010], metode inferensi yang dikembangkan ini dapat bekerja secara langsung pada basis pengetahuan Ternary Grid tanpa harus mengalami proses konversi dari basis aturan ke basis Ternary Grid. Dibandingkan dengan metode yang dikembangkan oleh [Yuliadi, 2010], metode yang dikembangkan ini bekerja lebih efisien. Pendekatan pemrograman rekursif yang diimplementasikan pada algoritma yang dikembangkan ini telah dapat mereduksi jumlah pengulangan program secara signifikan. Data-data pada tabel 1 berikut diambil dari beberapa percobaan:

Tabel 1. Data percobaan

Jumlah Aturan	Jumlah Fakta	Jumlah Pengulangan
10	35	75
20	70	170
30	100	347
50	180	756

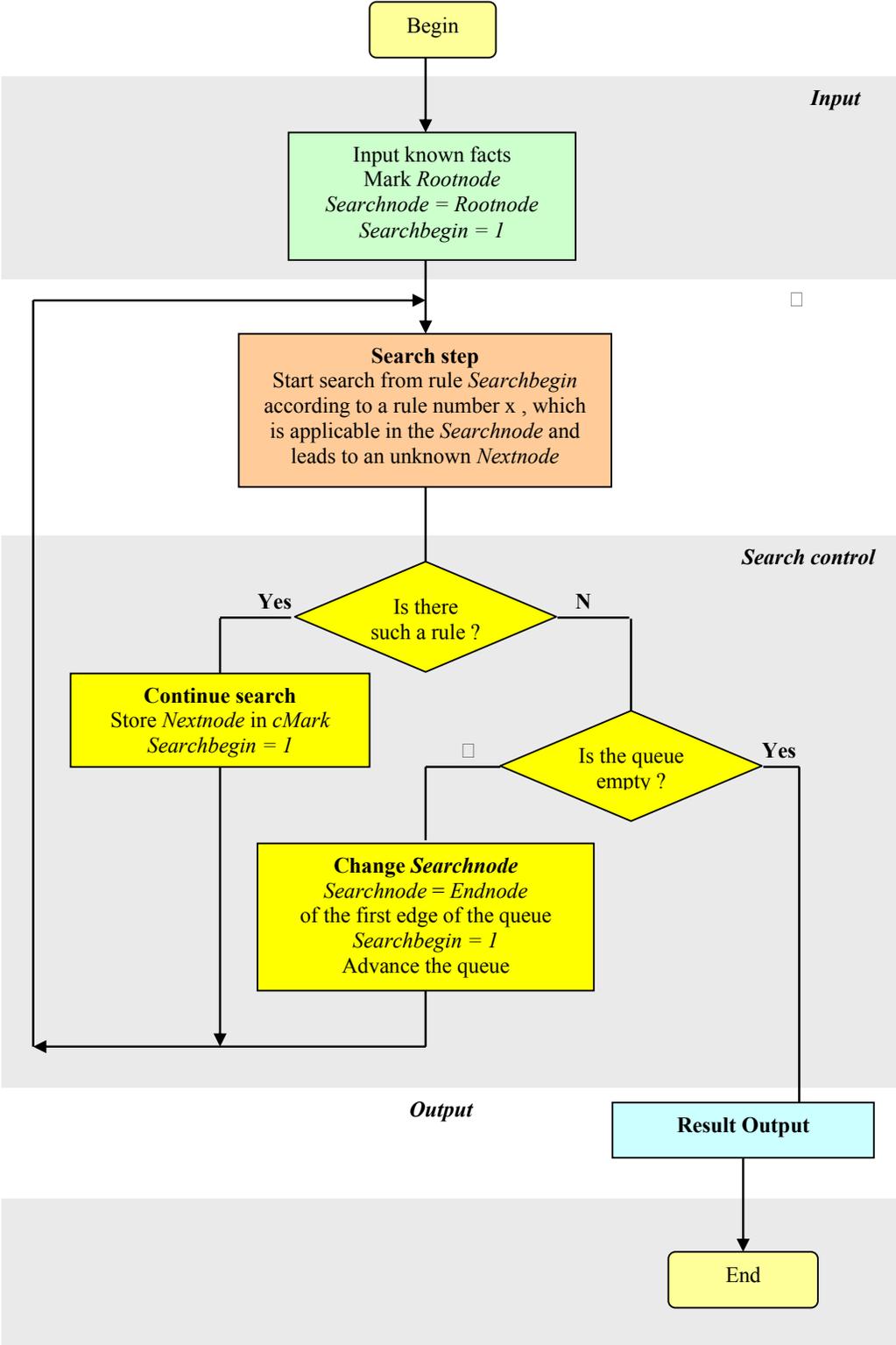
Gambar 9 menunjukkan kebutuhan proses rekursi yang disebabkan akibat peningkatan jumlah fakta dan aturan.



Gambar 9. Kebutuhan proses rekursi

Metode inferensi dengan forward chaining melalui pendekatan rekursif yang bekerja pada ternary grid ini berhasil dipublikasikan pada jurnal [Yuliadi, 2011]. Dengan demikian metode ini telah memperkaya metode-metode reasoning pada sistem pakar yang ada, khususnya yang berbasis ternary grid.

Agar proses inferensi secara forward chaining ini dapat dipahami dengan mudah dan jelas, berikut ini proses inferensi akan ditunjukkan dengan flow chart (gambar 10). Pada proses inferensi tersebut nampak 4 blok kegiatan utama, yaitu proses pembacaan masukan (input), proses pencarian (search), proses pengendalian (search control) dan proses hasil (output).



Gambar 10. Proses inferensi dengan forward chaining

## 5 METODOLOGI PELAKSANAAN

Pengembangan metode inferensi berbasis Ternary Grid ini akan diaplikasikan pada mesin inferensi dari sistem pakar. Mesin inferensi adalah program komputer yang berfungsi untuk menghasilkan jawaban dari basis pengetahuan atas pertanyaan pengguna. Mesin inferensi merupakan otaknya sistem basis pengetahuan yang mengolah informasi dari basis pengetahuan. Cara kerja dari mesin inferensi adalah mengolahnya fakta dan aturan yang disimpan pada basis pengetahuan. Terdapat dua teknik yang digunakan dalam melakukan inferensi, yaitu *forward chaining* dan *backward chaining* [Lunze, 1994].

Inferensi adalah proses membuat keputusan berdasarkan pengetahuan yang ada [David, 2003]. Bagi seorang pakar, inferensi berarti proses membuat keputusan atau memberikan informasi konsultasi berdasarkan keahlian yang dimilikinya. Perpaduan antara orang pakar dengan komputer menghasilkan apa yang disebut dengan sistem pakar. Sistem pakar dapat diartikan sebagai sekumpulan program komputer (perangkat lunak) yang bekerja seperti seorang pakar manusia dan berfungsi untuk menyelesaikan suatu permasalahan berdasarkan pengetahuan yang dimilikinya dan disimpan di memorinya (basis pengetahuan) [Patterson, 1990] [Tanimoto, 1990] [Williamson, 1986]. Salah satu komponen yang ada pada sistem pakar adalah mesin inferensi. Pada sistem pakar berbasis aturan, mesin inferensi bekerja mengolah fakta dan aturan untuk menentukan aturan mana yang dapat dipakai. Terdapat dua teknik yang digunakan dalam melakukan inferensi, yaitu *forward chaining* dan *backward chaining* [Lunze, 1994] [Giarratano, 2005], [Ignizio, 1991], [Jackson, 1998]. Meskipun metoda tersebut merupakan metoda baku, namun tidak menjamin akan efisiensi dan keefektifan suatu sistem pakar karena hal itu tergantung pula kepada struktur dan model pengetahuan yang diterapkan.

### 5.1 *Backward Chaining*

Kebalikan dengan *forward chaining*, pada *backward chaining*, hal yang pertama ditentukan adalah targetnya terlebih dahulu, yaitu fakta yang merupakan bagian simpulan dari suatu aturan. Berangkat dari target tersebut, mesin inferensi mencari aturan-aturan yang fakta bagian kondisinya memenuhi kriteria sesuai dengan himpunan fakta masukan (bergerak ke belakang). Apabila fakta-fakta bagian kondisi tersebut memenuhi kriteria, maka aturan tersebut dapat digunakan. Dari fakta-fakta baru yang didapat, selanjutnya mesin inferensi mencari lagi fakta-fakta lainnya dengan bergerak ke belakang, dari bagian simpulan ke bagian kondisi suatu aturan [Lunze, 1994] [Patterson, 1990] [James, 2008].

*Backward chaining* termasuk teknik inferensi yang efektif dan banyak diterapkan di berbagai aplikasi sistem pakar. *Backward chaining* disebut juga sebagai teknik inferensi yang dikendalikan oleh target (*goal driven*). Melihat contoh kasus yang diberikan pada gambar 2, mesin inferensi dengan *backward chaining* ini akan memulai dari target yang dimasukkan oleh pengguna. Misalnya fakta R diketahui, maka mesin inferensi menjadikan fakta R tersebut sebagai target. Selanjutnya mesin inferensi memproses fakta-fakta bagian kondisi dari aturan yang ada R tersebut. Jika fakta-fakta S, T dan U diketahui, maka aturan tersebut dapat digunakan dan mesin inferensi menyimpulkan bahwa fakta R memenuhi kriteria. Demikian proses inferensi

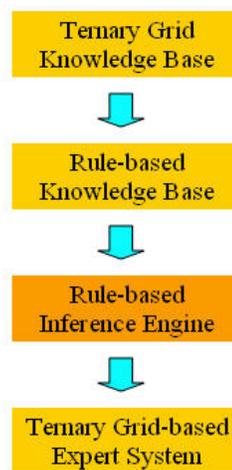
berlanjut hingga semua aturan yang memungkinkan untuk digunakan selesai diproses. Keuntungan dari backward chaining ini adalah dapat membuat kesimpulan-kesimpulan baru meskipun fakta-fakta yang ada tidak memenuhi kriteria, karena mesin akan memproses terus fakta-fakta yang ada ke belakang (ke bagian kondisi suatu aturan).

Meskipun backward chaining terbilang efektif dan banyak digunakan pada berbagai sistem pakar, namun [James, 2008] menyatakan bahwa penggunaan penggunaan teknik inferensi harus disesuaikan dengan permasalahan yang akan diselesaikan. Perbandingan antara bagian kondisi dan bagian simpulan dari aturan dapat dijadikan bahan untuk pemilihan teknik inferensi. Jika dari aturan-aturan yang ada secara rata-rata memiliki lebih banyak bagian kondisinya daripada bagian simpulannya, maka setiap hipotesis atau target akan menghasilkan banyak aktifitas untuk memproses kondisi suatu aturan, sehingga teknik inferensi dengan forward chaining pada situasi ini lebih sesuai. Jika situasinya sebaliknya, maka backward chaining lebih sesuai.

Sistem pakar yang dikembangkan oleh [Yuliadi, 2003] menggunakan teknik forward chaining dalam proses inferensinya. Disamping perbandingan bagian kondisi lebih banyak daripada bagian simpulannya dari aturan-aturannya yang ada, juga fungsi dari sistem pakar tersebut lebih kearah pembuatan keputusan pakar daripada melakukan validasi.

## 5.2 Metode

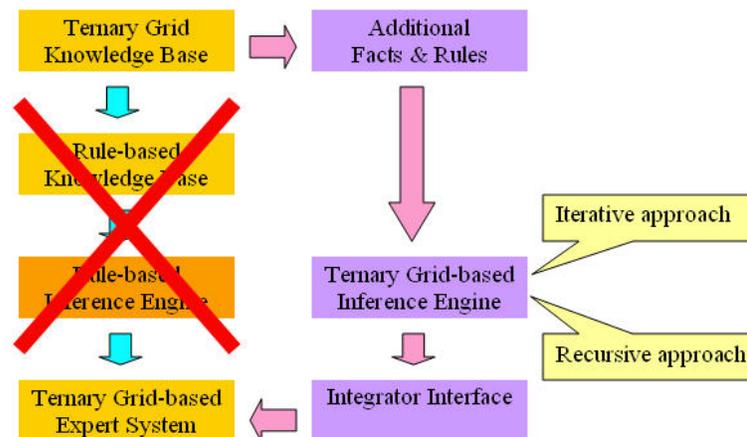
Sebagai bagian dari sistem pakar, Mesin Inferensi (inference engine) bertugas melakukan inferensi atau *reasoning* terhadap aturan-aturan yang disimpan pada Basis Pengetahuan. Mesin inferensi merupakan otaknya sistim basis pengetahuan yang mengolah informasi dari basis pengetahuan. Cara kerja dari mesin inferensi adalah mengolahnya fakta yang diberikan oleh pengguna dan mencari keterkaitan antara fakta-fakta tersebut dengan fakta-fakta dan aturan-aturan yang disimpan pada basis pengetahuan. Terdapat dua teknik yang digunakan dalam melakukan inferensi, yaitu *forward chaining* dan *backward chaining* [Lunze, 1994].



Gambar 11. Kondisi awal (saat ini)

Basis pengetahuan yang digunakan oleh Sistem Pakar pada penelitian ini memiliki format aturan produksi [Newell, 1972] yang komposisinya terdiri dari Fakta dan Aturan. Format tersebut telah dioptimalkan pada penelitian terdahulu [Erdani, 2005a] [Erdani, 2005b] [Erdani, 2005c] [Yuliadi, 2007c] agar didapatkan kinerja pengetahuan yang lebih baik. Format tersebut selanjutnya disebut sebagai Basis Pengetahuan Ternary Grid.

Kondisi saat ini yang terjadi pada sistem pakar berbasis Ternary Grid yaitu terdapat proses untuk mengkonversi balik basis pengetahuan dari model Ternary Grid ke model aturan agar aturan tersebut dapat digunakan oleh mesin inferensi berbasis aturan (gambar 2). Hasil dari penelitian ini diharapkan dapat mempersingkat proses yang dimaksud dan mengeliminasi dengan mesin inferensi baru yang mampu beradaptasi dengan basis pengetahuan berbasis Ternary Grid, sehingga terjadi peningkatan efisiensi pada proses keseluruhan sistem pakar (gambar 12).



Gambar 12. Kondisi akhir (hasil penelitian)

Dengan memperhatikan sifat-sifat dan karakter Ternary Grid, kami telah merumuskan tahapan untuk mengembangkan metode inferensi yang terdiri dari:

- Pengembangan dan penyempurnaan database facts & rules
- Pengembangan metode inferensi dengan pendekatan iterativ
- Pengembangan metode inferensi dengan pendekatan rekursiv
- Pengembangan software simulator inferensi
- Pengembangan software integrator interface

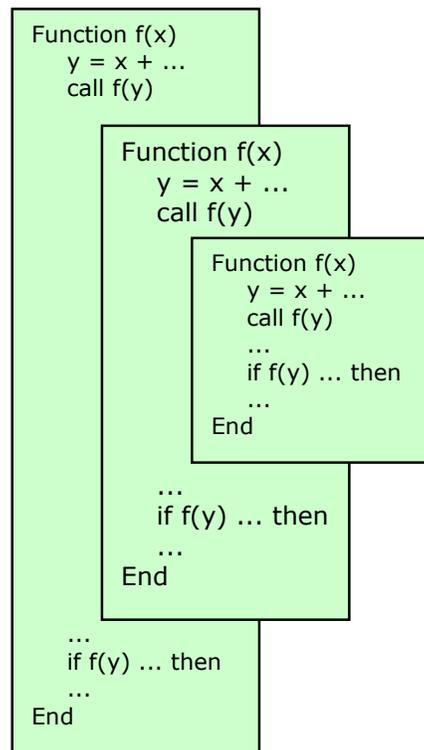
Pada tahun pertama Tim Peneliti akan melakukan studi literatur mengenai Ternary Grid, model pengetahuan, metode/teknik Inferensi, konsep simulator dan konsep interface. Selanjutnya Tim Peneliti akan mengembangkan dan melengkapi database facts & rules berbasis Ternary Grid yang prototype-nya sudah dikembangkan oleh [Erdani, 2005a] [Erdani, 2005b] [Erdani, 2005c] dan [Yuliadi, 2008b], dan menyempurnakan metode inferensi dengan pendekatan iterativ yang konsepnya pernah dikembangkan oleh [Yuliadi, 2008b]. Hasil dari tahap ini yaitu berupa basis

pengetahuan Ternary Grid yang sudah disempurnakan dalam bentuk Database dan dilengkapi dengan data-data pendukung proses inferensi serta metode inferensi sistem pakar dengan pendekatan iterative yang sudah disempurnakan serta siap diimplementasikan.

Pada tahun kedua Tim Peneliti akan melakukan pengembangan metode inferensi sistem pakar dengan pendekatan rekursif dan mengimplementasikannya ke dalam bentuk software simulator inferensi. Pendekatan rekursif ini diharapkan dapat lebih meningkatkan efisiensi proses inferensi.

Pendekatan rekursif yang dimaksud pada penelitian ini adalah pemrograman dengan metode rekursif atau dalam ilmu komputer dikenal dengan istilah *recursion*. Rekursif adalah metode dimana pemecahan masalah dilakukan dengan melokalisir area dimana masalah itu berada [Graham, 1990]. Menurut [Epp, 1995], pendekatan rekursif dapat diterapkan sebagai solusi berbagai permasalahan dan merupakan ide sentral dari ilmu komputer. Sementara menurut [Wirth, 1976]: *"The power of recursion evidently lies in the possibility of defining an infinite set of objects by a finite statement. In the same manner, an infinite number of computations can be described by a finite recursive program, even if this program contains no explicit repetitions."*

Rekursif dapat diilustrasikan sebagai berikut:



Gambar 13. Ilustrasi pemrograman rekursif

Pada beberapa bahasa program fungsional yang tidak memiliki instruksi pengulangan (*looping*), pemrograman rekursif dapat mensubsitusi instruksi pengulangan dengan hasil yang sama. Contoh berikut adalah bagaimana fungsi faktorial dapat diselesaikan baik dengan pemrograman iterativ maupun rekursif.

Tergantung dari algoritma yang dikembangkan, pemrograman rekursif dapat lebih efisien atau kurang efisien dibandingkan dengan pemrograman iteratif. Contoh berikut adalah sebuah contoh penyelesaian perhitungan faktorial yang dilakukan dengan metode iteratif dan rekursif. Dengan metode iteratif, faktorial dapat diselesaikan dengan rumus berikut:

$$\begin{aligned}
 f(n) &= n! \\
 &= n \cdot (n-1) \cdot (n-2) \cdot \dots \cdot 2 \cdot 1 \\
 &= \prod_{i=1}^n i
 \end{aligned}
 \tag{1}$$

Dalam notasi algoritma, persamaan tersebut dijelaskan sebagai berikut:

```

Function f(n)
  y = 1
  For i = 1 to n Do
    y = y*i
  return y
End
  
```

Jika  $n = 5$ , hasil dari algoritma tersebut dapat dilihat pada tabel 2 berikut:

Tabel 2. Perhitungan faktorial dengan metode iteratif

i	y	
		1
1	1 x 1	1
2	1 x 2	2
3	2 x 3	6
4	6 x 4	24
5	24 x 5	120

Dengan metode rekursif, persamaan matematika faktorial menjadi sebagai berikut:

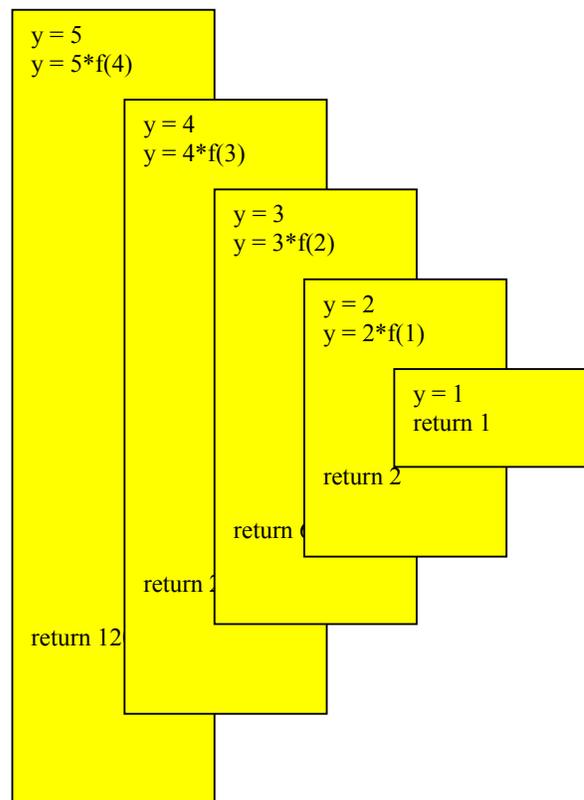
$$\begin{aligned}
 f(n) &= n! \\
 &= n \cdot (n-1) \cdot (n-2) \cdot \dots \cdot 2 \cdot 1 \\
 &= n \cdot f(n-1)! \quad \{n \in \mathbb{N} \mid n \geq 2\}
 \end{aligned}
 \tag{2}$$

Dalam notasi algoritma, persamaan tersebut dijelaskan sebagai berikut:

```

Function f(n)
  y = n
  If (n >= 2) Then
    y = y*f(n-1)
  return y
End
  
```

Jika n = 5, hasil dari algoritma tersebut dapat dilihat pada gambar 14 berikut:



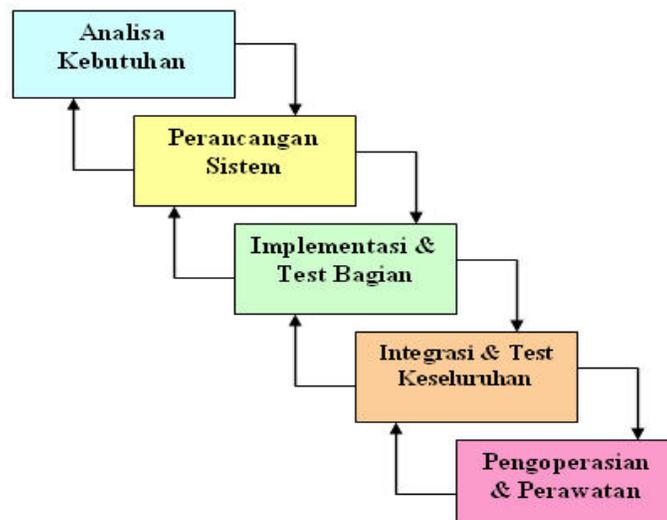
Gambar 14. Ilustrasi penyelesaian faktorial dengan rekursif untuk n=5

Dengan pendekatan rekursif yang sudah digambarkan tersebut, tim peneliti pada tahun ke 2 ini berusaha untuk mengimplementasikan pendekatan rekursif untuk menyelesaikan permasalahan yang ada pada mesin inferensi berbasis Ternary Grid.

Selanjutnya pada tahun terakhir (ketiga) Tim Peneliti akan melakukan pengembangan *integrator interface* yang dapat mengintegrasikan dan mensinkronkan komponen-komponen sistem pakar lainnya yang sudah dikembangkan pada penelitian terdahulu. Keseluruhan tahapan pekerjaan 3 tahun tersebut dapat dilihat pada sub bagian jadwal kerja (tabel 12).

Dalam pengembangan perangkat lunak, metoda yang digunakan mengacu kepada [Sommerville, 2001], dimana dinyatakan bahwa beberapa hal yang harus dipenuhi oleh software agar memiliki kriteria yang baik adalah kegunaannya (*usability*) dan kemampuan untuk mengantisipasi perubahan (*maintainability*). Demikianpun dengan software basis pengetahuan yang dikembangkan di dalam penelitian ini diharapkan dapat memenuhi kriteri software yang baik tersebut. Untuk itu metodologi yang digunakan di dalam penelitian ini harus dapat menghantarkan proses pembuatan software basis pengetahuan menjadi software yang baik.

Metode yang digunakan untuk mengembangkan software basis pengetahuan ini yaitu menggunakan Model Proses Air Terjun (*Water Fall Process Model*) [Sommerville, 2001] dimana di dalamnya secara garis besar berisi aktifitas pengumpulan dan analisa informasi dari berbagai literature dan melakukan perancangan, implementasi serta uji coba perangkat keras dan lunak.



Gambar 15. Proses pengembangan perangkat lunak model air terjun

Langkah-langkah yang dilakukan mengikuti Model Proses Air Terjun (*Water Fall Process Model*) [Sommerville, 2001], yang terdiri dari:

- Tahap 1: Analisa kebutuhan
- Tahap 2: Perancangan system
- Tahap 3: Implementasi dan tes bagian
- Tahap 4: Integrasi dan tes keseluruhan
- Tahap 5: Pengoperasian dan perawatan

Tahap 1 berisi pengumpulan kebutuhan secara lengkap, kemudian menganalisanya dan mendefinisikan kebutuhan yang harus dipenuhi oleh program yang akan dibangun. Tahap 2 berisi perancangan sistem berdasarkan hasil tahap 1. Pada tahap 3 desain rangkaian dan/atau program diterjemahkan ke dalam hardware atau kode-kode dengan menggunakan bahasa pemrograman yang sudah ditentukan. Program yang dibangun langsung diuji secara unit. Pada tahap 4 unit-unit hardware atau program software disatukan, kemudian diuji secara keseluruhan (system testing). Pada tahap 5 (tahap akhir) sistem yang dikembangkan dioperasikan dilingkungannya dan dilakukan pemeliharaan, seperti penyesuaian atau perubahan karena adaptasi dengan situasi yang sebenarnya. Gambar 6 menunjukkan tahapan yang akan dilakukan pada pengembangan aplikasi software ini.

Dalam analisa kebutuhan ini, beberapa aplikasi-aplikasi yang sudah dikembangkan dijadikan bahan kajian, dievaluasi dan dibandingkan, seperti basis pengetahuan pada sistem pakar berbasis CongaXpert yang pada awalnya dikembangkan dengan software komersial dari Microsoft yaitu MS Access, kemudian dikembangkan lanjut dengan software open source database MySQL.

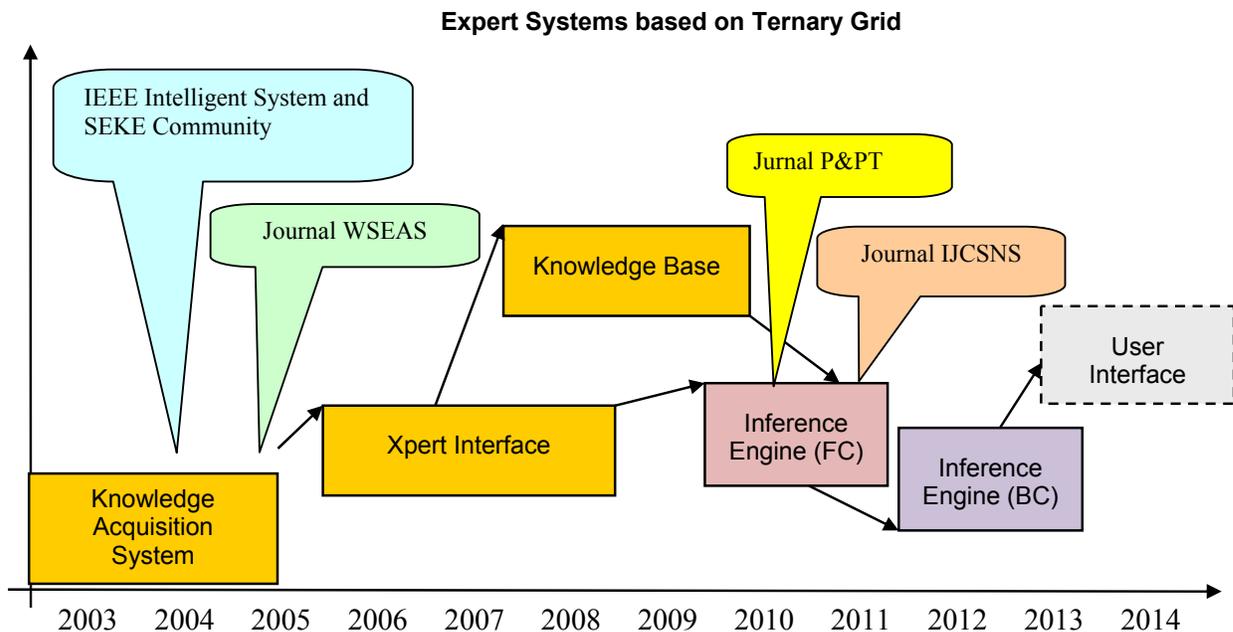
Analisa kebutuhan untuk software basis pengetahuan ini difokuskan kepada dua hal, yaitu:

- Kebutuhan fungsi
- Kebutuhan kualitas

Keduanya sangat sesuai dengan software-software yang akan dikembangkan.

### 5.3 Roadmap

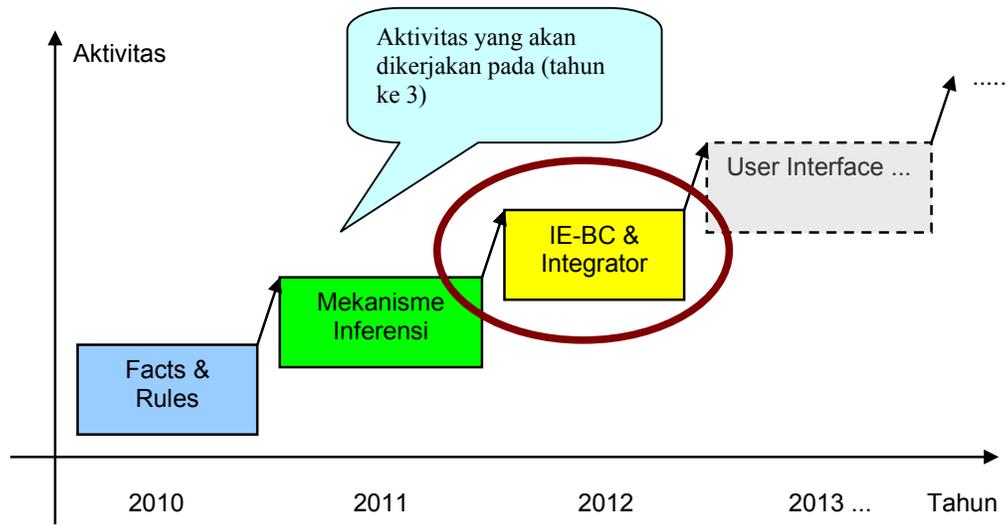
Roadmap penelitian merupakan pemetaan rencana riset yang berisi serangkaian proses perencanaan riset yang didorong oleh proyeksi/gambaran kebutuhan-kebutuhan (projected needs) atas kondisi masa yang akan datang dalam lingkup/domain kerja organisasi atau konteks tertentu.



Gambar 16. Roadmap lengkap pengembangan Sistem Pakar berbasis Ternary Grid

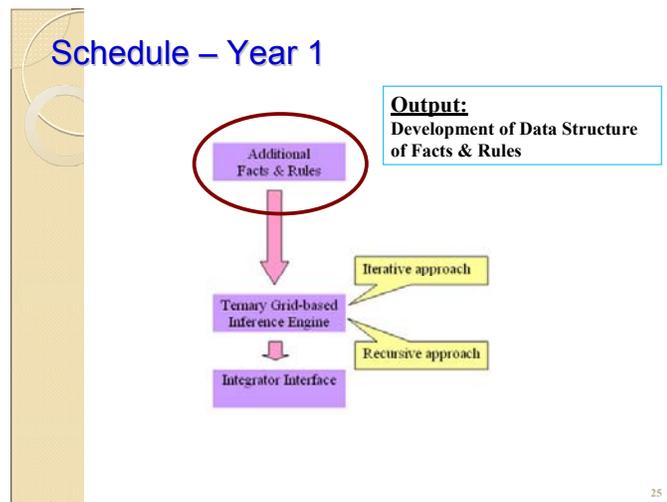
Roadmap selengkapnya terkait dengan pengembangan Sistem Pakar berbasis Ternary Grid terlihat pada gambar 16, sementara roadmap penelitian yang disusun pada skim Hibah Kompetensi ini terlihat pada gambar 8 dan berisi pemetaan kegiatan penelitian yang akan dilakukan selama 3 tahun, yang terdiri dari:

1. Pengembangan Struktur Data Facts & Rules
2. Pengembangan Mekanisme Inferensi dengan Pendekatan Iterativ dan Rekursiv
3. Pengembangan Mekanisme dan Metode Integrasi

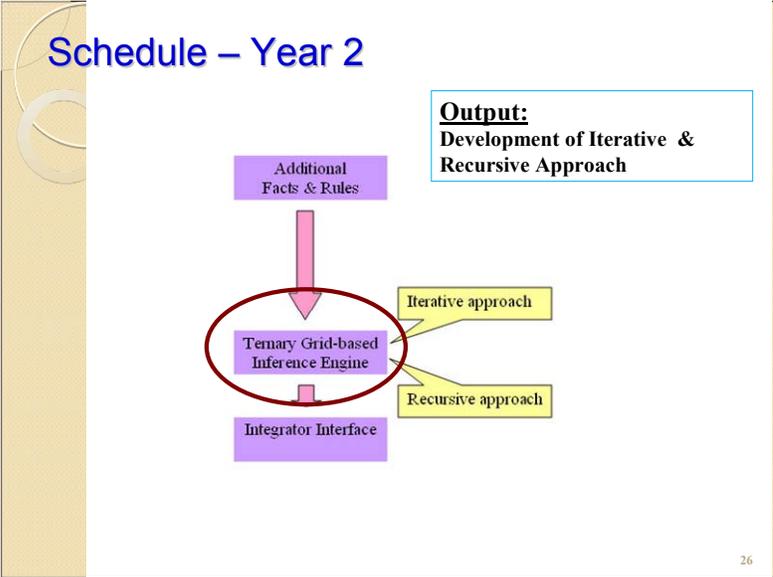


Gambar 17. Roadmap rinci skim Hibah Kompetensi

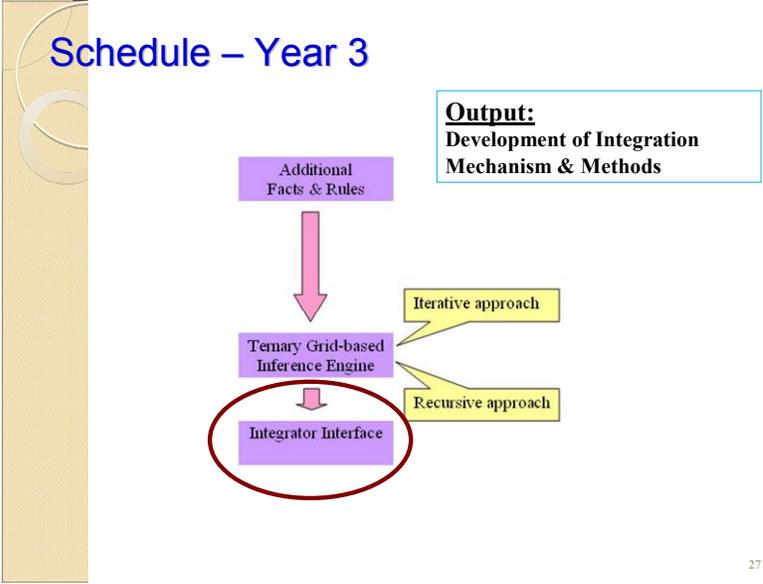
Gambar 18 sampai dengan gambar 20 berikut menunjukkan perincian kegiatan penelitian dari tahun pertama (2010) hingga tahun ke tiga (2012) yang mengacu kepada roadmap penelitian.



Gambar 18. Rencana kerja tahun I



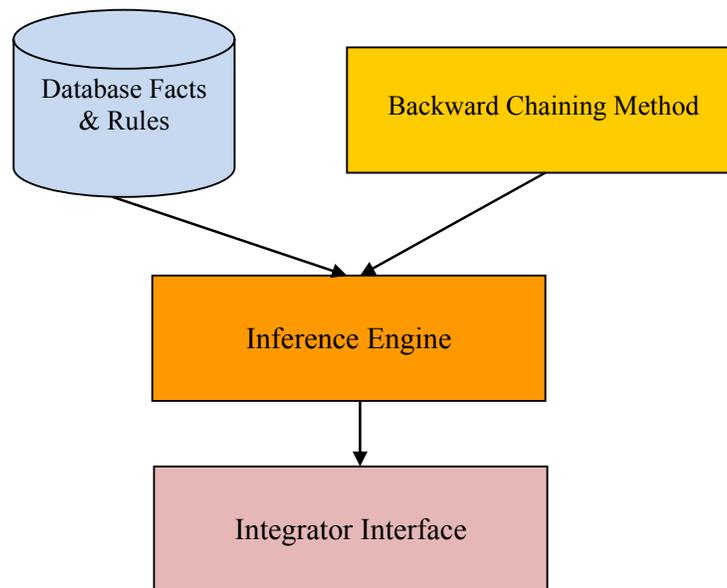
Gambar 19. Rencana kerja tahun II



Gambar 20. Rencana kerja tahun III

## 6 HASIL KEGIATAN TAHUN KE 3

Berdasarkan metodologi yang dikembangkan dan hasil dari penelitian terdahulu, maka untuk mengimplementasikan metode backward chaining pada mesin inferensi, ilustrasi diusulkan pada kegiatan penelitian tahun ke 3 sebagai berikut (gambar 21):



Gambar 21. Ilustrasi kegiatan tahun ke 3

Langkah kerja untuk mengimplementasikan ilustrasi tersebut disusun sebagai berikut:

- Membuat dan mengembangkan konsep backward chaining yang mampu bekerja pada ternary grid. Pada tahap ini dikembangkan algoritma yang berbasis rekursif untuk breath atau deep first search
- Menetapkan kebutuhan fakta dan aturan tambahan yang akan dimasukkan ke knowledge base. Fakta-fakta dan aturan-aturan tersebut merupakan tambahan yang berfungsi untuk mendukung dan menunjang proses inferensi.
- Memodifikasi knowledge base sesuai kebutuhan tersebut. Modifikasi dilakukan dalam bentuk penambahan data, dan memungkinkan perubahan struktur entity relationship model (ERM).
- Mengimplementasikan rancangan backward chaining dengan software. Pada tahap ini dikembangkan software GUI dengan VB 6.0 yang dapat berfungsi melakukan inferensi dengan backward chaining.
- Merancang dan mengimplementasikan integrator interface yang mengintegrasikan database dan inference engine baru dengan metode backward chaining. Integrator dibuat sedemikian rupa agar mudah dipahami baik oleh user maupun oleh expert.

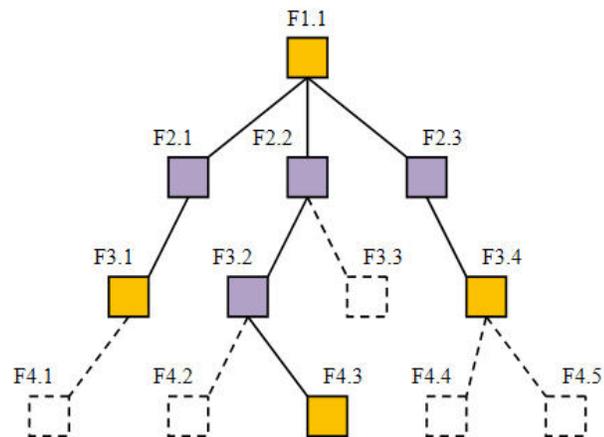
Mesin inferensi mengandung mekanisme pola pikir dan penalaran yang digunakan oleh pakar dalam menyelesaikan suatu masalah. Mesin inferensi adalah program komputer yang memberikan metodologi untuk penalaran tentang informasi yang ada dalam basis pengetahuan dan dalam workplace, dan untuk memformulasikan kesimpulan.

Mekanisme inferensi adalah bagian dari sistem pakar yang melakukan penalaran dengan menggunakan isi daftar aturan berdasarkan aturan dan pola tertentu. Selama proses konsultasi antar sistem dan pemakai, mekanisme inferensi menguji aturan satu demi satu sampai kondisi aturan tersebut benar.

Meskipun Ternary Grid sudah diaplikasikan pada sebuah mesin inferensi, tetapi pendekatan yang digunakannya masih dengan metode forward chaining. Metode *backward chaining* masih belum digunakan pada mesin inferensi berbasis Ternary Grid. Metode backward chaining yang dikembangkan pada penelitian ini diaplikasikan pada mesin inferensi sistem pakar berbasis Ternary Grid. Mesin inferensi tersebut diharapkan mampu mengolah semua informasi di dalam basis pengetahuan melalui pengaplikasian fakta dan aturan atau istilah dalam bahasa asing yang dikenal dengan nama firing facts and rules.

Backward chaining adalah strategi inferensi yang berlawanan dengan forward chaining. Strategi backward chaining dimulai dari tujuan (goal) dan berakhir pada fakta yang mengarah ke tujuan. Metode backward chaining dikenal pula dengan istilah *goal driven strategy of inference engine*. Pada beberapa literatur dikatakan bahwa proses inferensi dengan backward chaining dimulai dari elemen terakhir di dalam rantai menuju element pertama pada rantai. Proses ini seringkali sangat efektif, terutama pada sekuen yang kompleks pada proses inferensi.

Terdapat dua algoritma pencarian yang bisanya diaplikasikan pada metode backward chaining, yaitu algoritma depth-first dan breath-first search. Kedua algoritma tersebut melakukan pencarian data pada struktur data pohon (tree). Depth First Search adalah algoritma pencarian pada sebuah pohon atau tree. Pencarian DFS ini adalah dengan menelusuri satu cabang sebuah pohon sampai kebawah (menemukan solusi) sebelum melakukan backtracking. Breadth-first search adalah algoritma yang melakukan pencarian secara melebar yang mengunjungi simpul secara preorder yaitu mengunjungi suatu simpul kemudian mengunjungi semua simpul yang bertetangga dengan simpul tersebut terlebih dahulu. Selanjutnya simpul yang belum dikunjungi dan bertetangga dengan simpul-simpul yang tadi dikunjungi, demikian seterusnya. Jika graf berbentuk pohon berakar, maka semua simpul pada level  $x$  dikunjungi lebih dahulu sebelum simpul-simpul pada level  $x+1$ . Proses backward chaining menggunakan algoritma depth-first search dijelaskan pada ilustrasi berikut:



Gambar 22. Ilustrasi backward chaining pada struktur pohon

Gambar 22 menunjukkan proses inferensi sistem pakar dengan strateg backward chaining pada struktur pohon. Fakta yang sudah diketahui adalah F1.1, F3.1, F3.4 dan F4.3. Fakta F1.1. sekaligus sebagai goal. Proses inferensi dimulai dari F1.1. Mesin inferensi mencoba untuk mengaplikasikan F2.1, F2.2 dan F2.3. Langkah-langkah proses inferensi dengan backward chaning dijelaskan seagai berikut:

- Goal: F1.1
- Fetch: F2.1 → unknown
- Fetch: F3.1 → known
- Rule (F3.1, F2.1) → fired
- Rule (F2.1, F1.1) → fired
- Fetch: F4.1 → unknown
- Fetch: F2.2 → unknown
- Fetch: F3.2 → unknown
- Fetch: F4.2 → unknown
- Fetch: F4.3 → known
- Rule (F4.3, F3.2) → fired
- Rule (F3.2, F2.2) → fired
- Rule (F2.2, F1.1) → fired
- ...
- Etc.

Proses inferensi akan terus berlanjut ke dasar pojon (daun) hingga seluruh fakta yang ada pada pohon sudah dites semua untuk diaplikasikan. Algoritma proses inferensi menggunakan backward chaning dijelaskan dalam notasi berikut:

*Backward Chaining (...)*

```

While Rule Number <= Existing Rules AND
Rule is not applicable Do
  IF Rule applicable Then
    Incrementing Number of Marking Node
    IF Not End of Rule Then
      Incrementing Number of Active Node
      Listing Active Node
      Backward Chaining (Next Node , ...)
    Else
      IF Number of Active Node > 0 Then
        First Element ← List of Active Node
        Decrementing Number of Active Node
        Backward Chaining (Search Node, ...)
  
```

Pada algoritma tersebut dijelaskan bahwa proses inferensi terus berlangsung selama nomor aturan lebih kecil daripada jumlah aturan yang ada dan terdapat aturan yang belum diaplikasikan. Jika sebuah aturan diaplikasikan (fired), maka proses inferens menuju proses pencarian fakta-fakta yang mengarah ke aturan yang diaplikasikan tersebut hingga tidak ditemukan lagi fakta-fakta. Jika sebuah aturan tidak dapat digunakan, maka program akan mencari aturan lain pada jalur lain yang memungkinkan digunakan. Proses akan terus berlanjut hingga semua fakta selesai dieksekusi.

Fakta dan aturan berikut seagai contoh sederhana untuk proses inferens menggunakan backward chaining

```

Rule 1: IF <tutorial score >= 60> AND <thesis draft is finished> THEN <tutorial result passed>;
Rule 2: IF <tutorial result passed> AND <presentation score greater than or equals 60> AND <comprehensive test passed> THEN <final project passes>;
Rule 3: IF <final project passes> AND <thesis revision on time> THEN <graduation on time>;
Rule 4: IF <final project passes > AND <thesis revision not on time > THEN <graduation postponed>
  
```

Sesuai dengan teknik akusisi Ternary Grid, aturan-aturan tersebut dimasukan terlebih dahulu ke basis pengetahuan. Sesuai dengan konsep Ternary Grid, format aturan tersebut dikonversi terlebih dahulu ke format Ternary Grid. Seluruh aturan yang memungkinkan diaplikasikan, dimasukan ke dalam himpunan  $R_x$ . Secara matematika dijelaskan sebagai berikut:

$$R_x = \{ p \mid p \rightarrow q, p \in F_k, p \in F, F_k \subset F \} \quad (1)$$

Seluruh aturan yang dapat diaplikasikan dimasukkan ke himpunan Ryn sebagai berikut:

$$R_{y_n} \subset R_x \quad (2)$$

Pada aplikasi software yang dikembangkan, aturan yang dikonversi ke dalam format Ternary Grid disimpan seperti terlihat pada gambar 23.

Rule\Fact	F 1 (24)	F 2 (25)	F 3 (26)	F 4 (27)	F 5 (28)	F 6 (29)	F 7 (30)	F 8 (31)	F 9 (32)	F 10 (33)
R 1 (6)	1	1	2							
R 2 (7)			1	1	1	2				
R 3 (8)						1	1		2	
R 4 (9)						1		1		2

Gambar 23. Aturan dalam format Ternary Grid

Sementara fakta-fakta yang dikenal atau dimasukkan oleh pengguna ditunjukkan pada gambar 24.

Gambar 24. Fakta-fakta yang dikenal (dimasukan oleh pengguna)

Aturan yang tidak konsisten akan terdeteksi dan tereliminasi sesuai dengan penjelasan matematik berikut:

$$B = \{i \mid a_{ij} = 3\} \quad (3)$$

Demikianpun dengan duplikasi aturan akan terdeteksi dan tereliminasi sesuai dengan penjelasan matematik berikut:

$$C = \bigcup_{b \in B} b \quad (4)$$

Hasil proses inferensi dtunjukkan pada gambar 25. Hasil tersebut menunjukkan keefektifan algoritma yang dkembangkan. Dibandingkan dengan metode yang sudah dikembangkan oleh [Hunger, 1999] [Yuliadi, 2010] [Yuliadi, 2011], Metode ini dapat bekerja langsung pada Ternary Grid tanpa harus mengalami proses konversi dari format produksi ke formar Ternary Grid. Dibandingkan dengan [Yuliadi, 2010] [Yuliadi, 2011], proses inferensi ini bekerja lebih dinamis dan proses komputasinya lebih efisien (berkurangnya proses iterasi). Hasil proses inferensi dapat menunjukkan beberapa fakta yang sebelumnya tidak diketahui menjadi dikenal dan mengarahkan kepada aturan yang dapat diaplikasikan, sehingga dari proses ini menghasilkan simpulan yang lebih banyak. Dengan demikian sistem pakar dapat memberkan informasi yang lebih banyak lagi kepada pengguna.



Gambar 25. Hasil proses inferensi dengan backward chaining

Di dalam proses inferensi dimungkinkan terdapat aturan-aturan yang dapat mengganggu jalannya proses inferensi tersebut. Salah satunya adalah aturan yang tidak konsisten. Agar proses inferensi dapat berjalan dengan baik dan lancar, maka aturan-aturan tersebut harus dieliminasi dengan rumus dan algoritma berikut:

- Mencari sekumpulan baris pada aturan-aturan, dimana nilai 3 muncul (find rows, in which value 3 appears):

$$B = \{ i \mid a_{ij} = 3 \} \quad (7)$$

- Menghilangkan aturan-aturan yang duplikasi (remove row duplication):

$$C = \bigcup_{b \in B} b \quad (8)$$

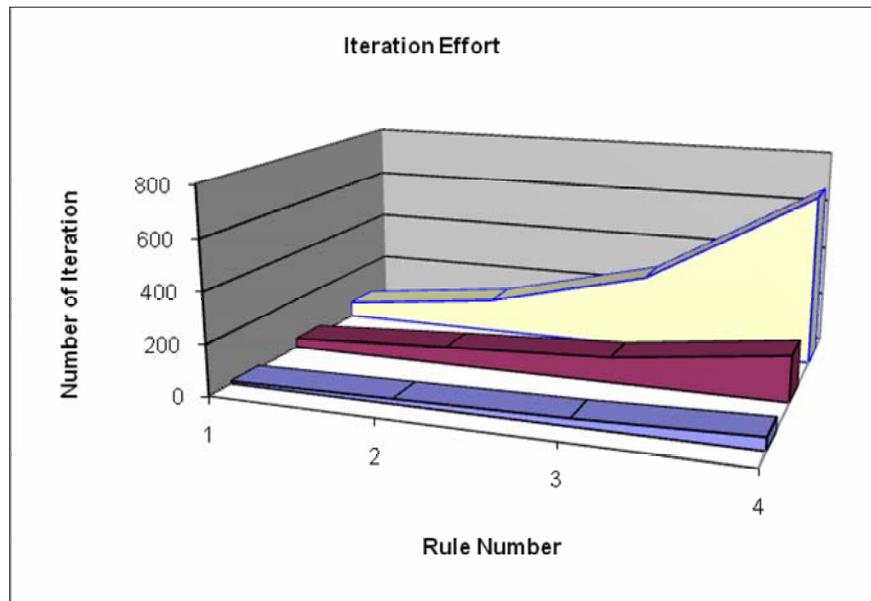
Proses inferensi dengan algoritma yang dikembangkan pada basis pengetahuan ternary grid ini [Yuliadi, 2012] menunjukkan bahwa algoritma yang dikembangkan di dalam penelitian ini bekerja cukup efektif. Dibandingkan dengan metoda yang sudah dikembangkan oleh [Hunger, 1999] [Yuliadi, 2010] dan [Yuliadi, 2011], metoda inferensi yang sudah dikembangkan dapat bekerja langsung pada basis pengetahuan ternary grid tanpa harus dilakukan proses konversi dari basis pengetahuan berbasis aturan (*rule-based knowledge*). Dalam hal metoda inferensi, dibandingkan dengan metoda-metoda inferensi yang sudah dikembangkan oleh [Yuliadi, 2010] dan [Yuliadi, 2011], metode inferensi yang sudah dikembangkan ini bekerja lebih dinamis dan efisien dalam proses komputasinya.

Pendekatan rekursif yang diimplementasikan pada mesin inferensi yang sudah dikembangkan ini mampu mereduksi jumlah proses pengulangan (looping) pada saat komputasi. Data-data berikut diambil dari beberapa eksperimen yang sudah dilakukan:

Tabel 2

Number of rule	Number of fact	Number of Loop
10	35	62
20	70	134
30	100	295
50	180	673

Ilustrasi grafis proses rekursi dapat dilihat pada gambar 26. Gambar tersebut menunjukkan bahwa jumlah rekursi yang terjadi sangat dipengaruhi oleh peningkatan jumlah fakta dan aturan, namun tidak proporsional.



Gambar 26. Data proses rekursi dalam tampilan grafis

## 7 KESIMPULAN

Mesin inferensi yang dikembangkan pada penelitian ini, yang berkerja dengan pendekatan metode backward chaining secara rekursif dapat berfungsi dengan baik. Mesin tersebut mampu menentukan fakta-fakta yang mengarahkan kepada aturan-aturan yang memungkinkan digunakan sehingga menghasilkan kesimpulan. Dibandingkan dengan metode sebelumnya yang menggunakan pendekatan iteratif atau forward chaining, metode yang dikembangkan ini dapat mereduksi jumlah pengulangan program secara signifikan sehingga dapat dikatakan lebih efisien. Merujuk kepada beberapa literatur terkait sistem pakar, metode yang dikembangkan ini adalah metode baru dan dapat memberikan kontribusi ilmiah bagi pengembangan sistem pakar.

### 8 ORGANISASI TIM PENELITI

Ketua Tim : Dr. Ing., Yuliadi Erdani, Dipl. EL. HTL, M.Sc.

Anggota Tim : Ridwan, SST.

: Setyawan Ajie Soekarno, SST

Pengorganisasian (susunan, tugas dan pembagian waktu) dan biodata Tim Peneliti selengkapnya ada pada lampiran 2.

### 9 JADWAL KEGIATAN

Tabel 3. Jadwal kegiatan

No.	Kegiatan	Tahun I				Tahun II				Tahun III			
		1	2	3	4	1	2	3	4	1	2	3	4
1	Studi literatur	X	X	X									
2	Perancangan dan pembuatan database facts & rules		X	X	X								
3	Pengembangan Metode Inferensi dengan interative approach		X	X	X								
4	Pengembangan Metode Inferensi dengan recursive approach					X	X	X					
5	Perancangan dan pembuatan software simulator inferensi						X	X	X				
6	Pengembangan konsep backward chaining									X	X		
7	Perancangan dan pembuatan software inferensi dengan backward chaining										X	X	
8	Perancangan dan pembuatan software integrator interface											X	X

## 10 PENDANAAN

### 10.1 Pendanaan Tahun ke 3 (2012)

Anggaran biaya tahun ke 3 (tiga) disusun pada tabel berikut:

Penggunaan Biaya	Biaya Satuan	Kuantitas	Jumlah
1. Honorarium			
a. Peneliti	Rp. 17.000.000	1	Rp. 17.000.000
<b>Sub Total 1</b>			<b>Rp. 17.000.000</b>
2. Pembuatan Alat/Software			
a. Additional Fact and Rule Database	Rp. 20.000.000	1	Rp. 10.000.000
b. Software Simulator Inferensi Backward Chaining + Integrator Interface	Rp. 25.000.000	1	Rp. 20.000.000
c. Software Integrator Interface	Rp. 15.000.000	1	Rp. 15.000.000
<b>Sub Total 2</b>			<b>Rp. 45.000.000</b>
3. Perjalanan dan/atau Publikasi			
a. Monev dan Pelaporan	Rp. 5.000.000	1	Rp. 5.000.000
<b>Sub Total 3</b>			<b>Rp. 5.000.000</b>
4. Operasionil			
a. Administrasi	Rp. 12.750.000	1	Rp. 12.750.000
b. Pajak Final	Rp. 4.250.000	1	Rp. 4.250.000
c. Operasionil	Rp. 1.000.000	1	Rp. 1.000.000
<b>Sub Total 4</b>			<b>Rp. 18.000.000</b>
<b>TOTAL</b>			<b>Rp. 85.000.000</b>

## **10.2 Justifikasi Pendanaan**

### **Honorarium**

Honorarium digunakan untuk membayar honor peneliti, anggota penelitian dan teknisi yang membantu penelitian ini.

### **Pembuatan Perangkat Lunak**

Kegiatan penelitian ini berisi pengembangan konsep yang diimplementasikan dalam bentuk pengembangan perangkat lunak. Mengingat kurangnya ketersediaan waktu dan tenaga yang dibutuhkan untuk pengembangan perangkat lunak, maka pengembangan perangkat lunak dilakukan melalui *out sourcing* dan *colaborative working* oleh pengembang software yang kompeten, berpengalaman dan terpercaya melalui suatu mekanisme seleksi kualitatif yang ketat. Pada tahun ke 3 pengembangan perangkat lunak berupa software simulator inferensi backward chaining dan integrator interface termasuk modifikasi database fakta dan aturan seharga total kurang lebih Rp. 45.000.000,-.

### **Perjalanan**

Biaya perjalanan diperlukan untuk menghadiri proses monev dan pelaporan.

### **Operasionil**

Biaya operasionil diperlukan untuk pendokumentasian dan pembuatan laporan, biaya publikasi pada jurnal atau prosiding, pelaksanaa administrasi dan biaya pajak.

## DAFTAR PUSTAKA

- [Erdani, 2003a] Erdani, Y., Hunger, A., Werner, Stefan., Mertens, S. Web-Based Consultation System with Expert System, *IATED CST 2003 – International conference (International Association of Science and Technology for Development – Computer Science and Technology)*, May 19-21, 2003, Cancun, Mexico, 2003 ISBN – 0-88986-349-0, page 61-64
- [Erdani, 2003b] Y. Erdani, A. Hunger, S. Werner, F.A Ekadiyanto: „Web-Based Distance Academic Consultation with Expert System”, ASEE/WFEO E-Conference 2003,
- [Erdani, 2004] Erdani, Y., Hunger, A., Werner, S., Mertens, S., Ternary Grid as a Potentially New Technique for Knowledge Elicitation/Acquisition, 2nd IEEE Conference on Intelligent System, vol I: pp. 312-315. ISBN 0-7803-8278-1, Varna - Bulgaria, June 22-24, 2004.  
This paper has been accepted also by SEKE 04 (Software Engineering and Knowledge Engineering) conference in Banff, Canada.
- [Erdani, 2005a] Erdani, Y., Hunger, A., Werner, S., Improving the Knowledge Performance using Ternary Grid Knowledge Acquisition and Model, *Proc. 4<sup>th</sup> WSEAS International Conference on Artificial Intelligence, Knowledge Engineering and Data Bases (AIKED 2005)*, Salzburg - Austria, 13-15 February 2005. ISBN 960-8457-09-2
- [Erdani, 2005b] Erdani, Y., Hunger, A., Werner, S., Improving the Knowledge Performance using Ternary Grid Knowledge Acquisition and Model, *WSEAS Transactions (Journals) on Information Science and Application, Issue 2, Volume 2*, February 2005. ISSN 1790-0832
- [Erdani, 2005c] Erdani, Y., Hunger, A., Werner, S., Improving the Knowledge Performance using Ternary Grid Knowledge Acquisition and Model, *Workshop WSB 2005 - Wissensbasierte Studienberatung im Internet zur Förderung der internationalen Wettbewerbsfähigkeit des Studienstandortes Deutschland*, 14-15 Februar 2005, Duisburg – Germany
- [Erdani, 2005d] Erdani, Yuliadi, *Acquisition of Human Expert Knowledge for Rule-based Knowledge-based Systems using Ternary Grid*, Verlag – Dissertation.de, Berlin, 2005, ISBN 3-89825-000-8
- [Yuliadi, 2006] Yuliadi Erdani, Tantangan dan Solusi Pengembangan GUI berbasis Web dengan Software Open Source, Proc. Seminar Nasional Strategi Masyarakat Open Source Software (OSS 2006) di Indonesia, 8 Agustus 2006, Bandung – Indonesia. ISBN 979-15035-0-8

- [Yuliadi, 2007a] Yuliadi Erdani, *The Role of Expert System in Education*, Abstract Proceeding Simposium Nasional 2007 - Puslitjaknov, 25-26 Juni 2007, Jakarta – Indonesia. ISBN 979-111-839-6
- [Yuliadi, 2007b] Yuliadi Erdani, *Konsep Pengembangan Sistem Pakar dengan Ternary Grid*, Proceeding Seminar Nasional Ilmu Komputer dan Aplikasinya (SNIKA 2007), Volume 2, No. 1, 1 November 2007, Bandung – Indonesia. ISSN 1907-882X
- [Yuliadi, 2007c] Yuliadi Erdani, *Mathematical Approach of Rule Optimization in Ternary Grid*, Proceeding. Seminar Nasional Ilmu Komputer dan Aplikasinya (SNIKA 2007), Volume 2, No. 1, 1 November 2007, Bandung – Indonesia. ISSN 1907-882X
- [Yuliadi, 2008a] Yuliadi Erdani, *Pengembangan Basis Pengetahuan (Knowledge Base) berbasis Ternary Grid*, Prosiding Seminar Nasional Informatika – Universitas Pembangunan Nasional “Veteran”, 24 Mei 2008, Yogyakarta, ISSN 1979-2328
- [Yuliadi, 2008b] Yuliadi Erdani, *Konsep Inferensi pada Model Pengetahuan berbasis Ternary Grid*, Prosiding Seminar Nasional Informatika – Universitas Pembangunan Nasional “Veteran”, 24 Mei 2008, Yogyakarta, ISSN 1979-2328
- [Yuliadi, 2009] Yuliadi Erdani, *Konsep Pengembangan Expert System Shell berbasis Ternary*, *Prosiding Seminar Nasional Industrial Service (SNIS 2009), Universitas Sultan Ageng Tirtayasa, Fakultas Teknik, Jurusan Teknik Industri, Pusdiklat PT. Krakatau Steel, 29 April 2009. ISBN 978-979-19280-0-7*
- [Yuliadi, 2010] Yuliadi Erdani, Pengembangan Metode Inferensi untuk Sistem Pakar berbasis Ternary Grid, *Jurnal P&PT, Asosiasi Politeknik Indonesia*, Desember 2010, ISSN 0854-5766
- [Yuliadi, 2011] Yuliadi Erdani, Developing Recursive Forward Chaining Method in Ternary Grid Expert Systems, *IJCSNS (International Journal of Computer Science and Network Security)*, Vol. 11 No. 8, August 2011, ISSN 1738-7906
- [Yuliadi, 2012] Yuliadi Erdani, Developing Backward Chaining Algorithm of Inference Engine in Ternary Grid Expert System, *IJACSA (International Journal of Advance Computer Science and Applications)*, Vol. 3 Issue 9, September 2012, ISSN 2156 5570 (online)
- [ACM-SIGCHI, 2008] ACM-SIGCHI (Association for Computing Machinery – Special Interest Group for Computer Human Interaction), accessed from the following website at Nov. 10th, 2008, [http://sigchi.org/cdg/cdg2.html#2\\_1](http://sigchi.org/cdg/cdg2.html#2_1)
- [David, 2003] David J.C. McKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.
- [Dix, 2004] Dix, Alan., et.al. (2004), *Human Computer Interaction*, Pearson Education, ISBN 0-13-046109-1

- [Feigenbaum,1993] Feigenbaum, E. A, Engelmores, R. S., Expert System and Artificial Intelligence, Japanese Technology Evaluation Center panel's report about Knowledge-Based Systems, Japan, 1993
- [Giarratano, 2005] Joseph C. Giarratano, Gary Riley Expert Systems, Principles and Programming, 2005, ISBN 0-534-38447-1
- [Green, 2008] Green, Paul (2008). Iterative Design. Lecture presented in Industrial and Operations Engineering 436 (Human Factors in Computer Systems, University of Michigan, Ann Arbor, MI, February 4, 2008.
- [Griffin, 2000] Griffin, E. (2000). A first look at communication theory (4th ed.). Boston, MA: McGraw-Hill. 393-404
- [Harmon, 1990] Harmon, P., Sawyer B., Creating Expert System for Business and Industry, John Wiley & Sons Inc, ISBN 0-471-61495-5, Canada, 1990
- [Herring, 2004] Herring, S. C. (2004). Computer-mediated discourse analysis: An approach to researching online behavior. In: S. A. Barab, R. Kling, and J. H. Gray (Eds.), Designing for Virtual Communities in the Service of Learning (pp. 338-376). New York: Cambridge University Press.
- [Hunger, 1999] Hunger, A., Werner, S., CONGA: A Course Online/ Offline Information and Guidance System to support an International Degree Course, Proceeding of ICCE 99 (Chiba-Japan, 1999, ISBN 1 58603 027 2, Page 577-583)
- [Ignizio, 1991] James Ignizio, Introduction to Expert Systems, 1991. ISBN 0-07-909785-5
- [Indah, 2009] Indah Kusumastuti, Yatri (2009). "Chapter 2: Komunikasi dalam Organisasi". Komunikasi Bisnis (edisi ke-edisi ke-1). IPB Press. ISBN 978-979-493-205-6.
- [Jackson, 1998] Peter Jackson Introduction to Expert Systems, 1998. ISBN 0-20-187686-8
- [Jain, 2008] Jain, M.B. Srinivas, M.B. ; Jain, A. A novel Web based Expert System Architecture for on-line and off-line fault diagnosis and control (FDC) of transformers. TENCON 2008 - 2008 IEEE Region 10 Conference. E-ISBN : 978-1-4244-2409-2
- [Lunze, 1994] Lunze, Prof. Dr.-Ing. Jan : Künstliche Intelligenz für Ingenieure. Band 1 : Methodische Grundlagen und Softwaretechnologie. Oldenbourg Verlag, München Wien, 1994, ISBN 3-486-22287-2
- [Markman, 2006] Markman, K. M. (2006). Computer-mediated conversation: The organization of talk in chat-based virtual team meetings. Dissertation Abstracts International, 67 (12A), 4388. (UMI No. 3244348)
- [McGraw, 1989] McGraw K.L., Harbison-Briggs K., Knowledge Acquisition for Expert System. Principles and Guidelines Prentice-Hall

- International Editions By: Karan.L. Mcgraw and Karan Harbison-Briggs, Page 1- 27, 1989
- [McQuail, 2005] McQuail, Denis. (2005). *Mcquail's Mass Communication Theory*. 5th ed. London: SAGE Publications
- [Newell, 1972] A. Newell and H.A. Simon, *Human problem solving* (Prentice Hall, Englewood Cliffs, NJ, 1972).
- [Patterson, 1990] Patterson, Dan W, *Artificial Intelligence and Expert System*, Prentice Hall, Englewood Cliffs, New Jersey, 1990, ISBN 0-13-477100-1
- [Preece, 1998] Preece, A. D., Borrowman A. J., Francis, T. J., *Reusable Components for Knowledge Base and Database Integration*, Technical Report AUCS/TR9803, 1998
- [Pressman, 2001] Pressman, Roger S. (2001). *Software engineering: a practitioner's approach*. ISBN 0073655783
- [Sascha, 2004] Sascha Mertens, Marius Rosu, Yuliadi Erdani, *An intelligent dialogue for online rule based expert systems*, Proc. of the 2004 International Conference on Intelligent User Interfaces, January 13-16, 2004, Funchal, Madeira, Portugal. ACM 2004, ISBN 1-58113-815-6
- [Sommerville, 2001] *Waterfall model in Software Engineering* by Ian Sommerville, Addison-Wesley, 2001
- [Schmidt, 2000] Schmidt, Michael .(2000). *Implementing the IEEE Software Eningeering Standard*. SAM Publishing. ISBN 0-672-31857-1
- [Smith, 1996] Smith, P., *An introduction to knowledge engineering*. London: International Thompson Computer Press, 1996.
- [Sutcliffe, 1995] Sutcliffe, A. G., *Human-Computer Interface Design*, 2nd ed. Houndsmills, Basingstoke, Hampshire: Macmillan Press Ltd, 1995.
- [Tsai, 1994] Tsai, N, Necco, C., Wei, G., *An assessment of current expert systems: Are your expectations realistic?* *Journal of Systems Management*, November 1994, 28-32, 1994
- [Thurlow, 2004] Thurlow, C., Lengel, L. & Tomic, A. (2004). *Computer mediated communication: Social interaction and the internet*. London: Sage
- [Verderber, 2005] Verderber, Rudolph F. (2005). "Chapter 4: Communicating through Nonverbal Behaviour". *Communicate!* (edisi ke-edisi ke-11). Wadsworth. ISBN 0-534-73936-4.
- [Wei-Su, 2001] Wei-Su, Kuo, et all. *A developed model of expert system interface (DMESI)*, Elsevier Science, 2001
- [Wilbert, 2007] Wilbert O. Galitz (2007), *the Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques*, Pub. John Wiley and Sons, ISBN 047-005342-9

**LAMPIRAN 2: SUSUNAN ORGANISASI**

No.	Nama	Jabatan dalam Tim	Uraian Tugas
	NIP/NIDN	Alokasi Waktu, Jam/Minggu	
1	Yuliadi Erdani	Ketua Tim	Membuat konsep dan rancangan keseluruhan dari antarmuka sistem pakar dan software aplikasi
	196807021997021001/0002076803	16 jam/minggu	
2	Ridwan	Anggota Tim	Mengembangkan basis data penunjang simulator inference engine sesuai dengan konsep yang dikembangkan oleh Ketua Tim
	1978 06 12 2001 12 1002	6 jam/minggu	
3	Setyawan Ajie Soekarno	Anggota Tim	Mengembangkan simulator inference engine sesuai dengan konsep yang dikembangkan oleh Ketua Tim
	1980 04 28 2008 10 1001	6 jam/minggu	

### LAMPIRAN 3: KETERSEDIAAN SARANA DAN PRASARANA PENELITIAN

Sarana dan prasarana yang tersedia untuk mendukung kegiatan penelitian ini terdiri dari:

- Laboratorium komputer dan informatika di Jurusan Otomasi Manufaktur dan Mekatronika
- Peralatan komputer dan laptop
- *Software development tools* Visual basic versi 6.0 ke atas
- Database MS Access dan SQL Server
- PHP Triad
- Java Applet dan Java Servlet *development kit*
- Fasilitas yang ada pada beberapa komunitas/pengembang software di kota Bandung dengan model kolaborasi kegiatan

**LAMPIRAN 4: BIODATA KETUA**

**I IDENTITAS DIRI**

1.1	Nama Lengkap (dengan gelar)	<b>Dr. Ing., Yuliadi Erdani, Dipl.Ing., M.Sc.</b>
1.2	Jabatan Fungsional	Lektor Kepala
1.3	NIP	1968 07 02 1997 02 1001
1.4	NIDN	0002076803
1.5	Tempat dan Tanggal Lahir	Bandung, 2 Juli 1968
1.6	Alamat Rumah	Sarijadi Blok 16 No. 103, Bandung – 40151
1.7	Nomor Telepon/Faks	022 – 201 91 97
1.8	Nomor HP	081 321 908 377
1.9	Alamat Kantor	Polman Bandung Jln. Kanayakan no. 21 – Dago, Bandung – 40135
1.10	Nomor Telepon/Faks	022-250 0241/ 022- 2649
1.11	Alamat e-mail	<a href="mailto:yul_erdani@yahoo.com">yul_erdani@yahoo.com</a>
1.12	Lulusan yg telah dihasilkan	D3= 60; S-1= 20 orang; S-2= 4 orang;
1.13	Mata Kuliah yg diampu	1. Algoritma dan Pemrograman (Inf. Tek. D4) 2. Informatika Teknik 3. Komunikasi Data 4. Program Aplikasi 5. Statistik 6. Pre Project Work

**II RIWAYAT PENDIDIKAN**

2.1 Program:	S-1	S-2	S-3
2.2 Nama PT	Ingenieurschule Burgdorf (ISB)/HTA Bern	Universitaet Duisburg, Germany	Universitaet Duisburg-Essen, Germany
2.3 Bidang Ilmu	Teknik Elektro	Computer Science and Communication Egineering (CSCE)	Informationstechnik
2.4 Tahun Masuk	1992	1999	2002
2.5. Tahun Lulus	1995	2002	2005
2.6 Judul Skripsi/ Tesis/Disertasi	<i>Entwicklung HGUE Modell</i>	<i>Usage an Expert System in a Web based Student Consultation System</i>	<i>Acquisition of Human Expert Knowledge for Rule-based Knowledge Based Systems using Ternary Grid</i>
2.7. Nama Pembim- bing/ Promotor	Prof. Hagemeyer	Prof. Dr. -Ing., Axel Hunger	Prof. Dr.-Ing., Axel Hunger

**III PENGALAMAN PENELITIAN (bukan skripsi, tesis, maupun disertasi)**

No.	Tahun	Judul Penelitian	Pendanaan	
			Sumber*	Jml (Juta Rp)
1	2010 – 2012	“Pengembangan Metode Inferensi Untuk Sistem Pakar Berbasis Ternary Grid”	Dikti-DP2M Penelitian Hibah Kompetensi	230
2	2007 – 2009	“Pengembangan Sistem Basis Pengetahuan (Sistem Pakar) Berbasis Ternary Grid”	Dikti-DP2M Penelitian Hibah Bersaing (HB)	150
3	2009	“Pengembangan Simulator Smart Traffic Light untuk Peningkatan Efektivitas Pengaturan Lalu Lintas di Perempatan Jalan”	Penelitian Strategis Nasional (DIPA PT)	100
4	2009	“Pengembangan Pelacak Lokasi Kendaraan dengan media GPS-GSM”	Research Grant I-MHERE	30
5	2008	“Pengembangan Sistem Kendali Nirkabel dan Kamera Pemantau Pada Robot Penjelajah”	Research Grant I-MHERE	30
6	2008	“Pengembangan Modul Praktikum Tele-Control via Jaringan Komunikasi Multi-Dropping dengan Mikrokontroler dan Visual Basic”	Research Grant I-MHERE	30
7	2007	“Pengembangan Jam Digital Tersebar (Distributed Digital Clock) dengan Jaringan Multidropping Untuk Acuan Waktu Kerja Standar POLMAN Bandung”	Research Grant I-MHERE	30
8	2006	“Pengembangan Jam Induk untuk Acuan Waktu Kerja Standar POLMAN Bandung”	DIPA PT	10

*Tuliskan sumber pendanaan: PDM, SKW, Fundamental Riset, Hibah Bersaing, Hibah Pekerti, Hibah Pascasarjana, RAPID, atau sumber lainnya.*

**IV PENGALAMAN PENGABDIAN KEPADA MASYARAKAT**

No.	Tahun	Judul Pengabdian Kepada Masyarakat	Pendanaan	
			Sumber*	Jml (Juta Rp)
1	2012	Penelaah/Evaluator Buku Ajar Politeknik Indramayu	SP4	

2	2012	Narasumber Pelatihan Metode Pembelajaran Teknik (MPT) untuk Staf Polindra, AK Karawang, AK Multistrada	SP4	
3	2012	Konsultasi Pengembangan GIO Traktor dan Water Treatment di SMK SMTI Pontianak	APBN-P	
4	2012	Narasumber Seminar Penelitian Terapan dan Pengabdian Pada Masyarakat untuk Peningkatan Daya Saing Daerah di Politeknik Cilacap	SP4	
5	2011	Detasering Ketua Prodi Teknik Informatika Politeknik Cilacap	DIKTI-SP4	
6	2010	Memberikan pelatihan kepada dosen Politeknik Aceh tentang aplikasi pemrograman komputer	USAID	
7	2009	Memberikan pelatihan kepada dosen Politeknik Negeri Ambon tentang penelitian dan penulisan artikel ilmiah	NUFFIC – MDF	
8	2009	Memberikan pelatihan kepada dosen Politeknik Aceh tentang database dan pemrograman komputer	USAID	
9	2008	Memberikan pelatihan kepada dosen Politeknik Aceh tentang dasar pemrograman komputer dan algoritma	USAID	
10	2007	Memberikan pelatihan kepada dosen Politeknik Kampar – Riau tentang Teknologi Informasi dan Komunikasi	Pemda Kampar – Riau	
11	2007	Memberikan penyuluhan kepada Masyarakat Mekatronika Indonesia untuk pengembangan Jurnal Ilmiah	Sukarela	
12	2007	Memberikan pelatihan kepada dosen Politeknik Padang – Sumbar tentang Mekatronika	Politeknik Padang – Sumbar	
13	2006	Memberikan pelatihan kepada Instruktur BLKI Serang – Banten Mekatronika	BLKI Serang – Banten	

*Tuliskan sumber pendanaan: Penerapan Ipteks, Vucer, Vucer Multitahun, UJI, Sibermas, atau sumber lainnya.*

#### V PENGALAMAN PENULISAN ARTIKEL ILMIAH DALAM JURNAL

No.	Tahun	Judul Artikel Ilmiah	Volume/ Nomor	Nama Jurnal
1	2005	“Improving the Knowledge Performance using Ternary Grid Knowledge Acquisition and Model”,	<i>Volume 2</i> February 2005. ISSN	<i>WSEAS Transactions on</i>

			1790-0832	<i>Information Science and Application, Issue,</i>
2	2007	“Human Expert Knowledge Acquisition Using Ternary Grid”	Edisi No. 3, Tahun XXI, September 2007, Jakarta – Indonesia. ISSN 0215-1685	Jurnal Teknologi (Journal of Technology) Fakultas Teknik Universitas Indonesia,
3	2010	Rancang Bangun Aplikasi Tele-Kendali Komputer Via Jaringan PSTN dengan Modul DTMF dan Mikrokontroler Attiny2313	Volume 01, Nomor 2, 2010 ISSN 2087-3379	Journal of Mechatronics, Electrical Power and Vehicular Technology (JMEV)
4	2010	Rancang Bangun Kendali Sekuen untuk Sambungan Jala-Jala Listrik Menggunakan Cycloconverter	Volume 01, Nomor 2, 2010 ISSN 2087-3379	Journal of Mechatronics, Electrical Power and Vehicular Technology (JMEV)
5	2010	Pengembangan Metode Inferensi untuk Sistem Pakar berbasis Ternary Grid	Vol. VIII, No. 2, Desember 2010, ISSN 0854-5766	Jurnal P&PT (Penerapan dan Pengembangan Teknologi)
6	2011	Developing Recursive Forward Chaining Method in Ternary Grid Expert Systems	Vol. 11, No. 8, August 2011, ISSN 1738-7906	IJCSNS (International Journal of Computer Science and Network Security)
7	2012	Developing Backward Chaining Algorithm of Inference Engine in Ternary Grid Expert System	Vol. 3, Issue 9, September 2012, ISSN 2156-5570 (Online), ISSN 2158-107X (Print)	IJACSA (International Journal of Advanced Computer Science and Applications)

8	2003	Web-Based Consultation System with Expert System,	Proc. <b>IASTED</b> CST 2003 – International conference (International Association of Science and Technology for Development – Computer Science and Technology), May 19-21, 2003, Cancun. Mexico. ISBN – 0-88986-349-0, page 61-64
9	2004	"Ternary Grid as a Potentially New Technique for Knowledge Elicitation/Acquisition"	Proc. 2nd <b>IEEE</b> Conference on Intelligent System, Varna - Bulgaria, June 22-24, 2004. vol I: pp. 312-315. ISBN 0-7803-8278-1
10	2004	An intelligent dialogue for online rule based expert systems	Proc. of the 2004 International Conference on Intelligent User Interfaces, January 13-16, 2004, Funchal, Madeira, Portugal. <b>ACM</b> 2004, ISBN 1-58113-815-6
11	2005	“Mathematical Approach for Solving Knowledge Optimisation Problem in Ternary Grid Knowledge Acquisition”,	accepted by <i>The 19<sup>th</sup> European Conference on Modelling and Simulation (ECMS) 2005 (formerly referred to ESM 2005)</i> , June 1-4, 2005, Riga – Latvia.
12	2006	“Developing an Expert-Interface for Knowledge Acquisition”,	Proc. Seminar Nasional Ilmu Komputer dan Aplikasinya (SNIKA 2006), vol. 1, no. 1, 9 November 2006, Bandung – Indonesia. ISSN 1907-882X
13	2006	“Organizing Factual Knowledge Efficiently”,	Proc. National Seminar on Knowledge Management (SKM 2006) – Knowledge Management and Competitive Value, Key Success Factor in Business, August 5 <sup>th</sup> & 7 <sup>th</sup> , 2006, Bandung – Indonesia. ISBN 979-15120-0-0
14	2006	“Developing Algorithms of Ternary Grid Technique for Optimizing Expert System Knowledge Base”,	Proc. Seminar Nasional Aplikasi Teknologi Informasi (SNATI 2006), June 17 <sup>th</sup> , 2006, Yogyakarta – Indonesia. ISSN 1907-5022
15	2006	“Applying Simply Web-Application and Expert System into PC-based Control System”,	Proc. <i>The 7<sup>th</sup> Seminar on Intelligent Technology and Its Applications (SITIA 2006)</i> , May 3 <sup>rd</sup> , 2006, Surabaya – Indonesia. ISBN-979-95989-8-2

16	2007	“Mathematical Approach of Rule Optimization in Ternary Grid”	Proceeding. Seminar Nasional Ilmu Komputer dan Aplikasinya (SNIKA 2007), Volume 2, No. 1, 1 November 2007, Bandung – Indonesia. ISSN 1907-882X
17	2007	“The Role of Expert System in Education”	Abstract Proceeding Simposium Nasional 2007 - Puslitjaknov, 25-26 Juni 2007, Jakarta – Indonesia. ISBN 979-111-839-6
18	2008	“Pengembangan Algoritma Inferensi dengan Pendekatan Iteratif pada Aturan berbasis Ternary Grid”	Prosiding Seminar on Application and Research in Industrial Technology (SMART), Universitas Gajah Mada, 27 Agustus 2008, Yogyakarta – Indonesia, ISBN 978-979-18528-0-7
19	2008	“Konsep Inferensi pada Model Pengetahuan berbasis Ternary Grid”	Prosiding Seminar Nasional Informatika, Universitas Pembangunan Nasional “Veteran”, 24 Mei 2008, Yogyakarta – Indonesia, ISBN 1979-2328
20	2008	“Konsep Basis Pengetahuan (Knowledge Base) berbasis Ternary Grid”	Prosiding Seminar Nasional Informatika, Universitas Pembangunan Nasional “Veteran”, 24 Mei 2008, Yogyakarta – Indonesia, ISBN 1979-2328
21	2009	“Konsep Penerapan Manajemen Pengetahuan di Lingkungan Polman Bandung”	Prosiding Seminar Nasional Industrial Service (SNIS 2009), Universitas Sultan Ageng Tirtayasa, Fakultas Teknik, Jurusan Teknik Industri, Pusdiklat PT. Krakatau Steel, 29 April 2009. ISBN 978-979-19280-0-7
22	2009	“Konsep Pengembangan Expert System Shell berbasis Ternary”,	Prosiding Seminar Nasional Industrial Service (SNIS 2009), Universitas Sultan Ageng Tirtayasa, Fakultas Teknik, Jurusan Teknik Industri, Pusdiklat PT. Krakatau Steel, 29 April 2009. ISBN 978-979-19280-0-7

**VI PENGALAMAN PENULISAN BUKU**

No.	Tahun	Judul Buku	Jumlah Halaman	Penerbit
1	2005	“Acquisition of Human Expert Knowledge for Rule-based Knowledge-based Systems using Ternary Grid”	150	Verlag – Dissertation.de, Berlin, 2005, ISBN 3-89825-000-8

**VII PENGALAMAN PEROLEHAN HKI**

No.	Tahun	Judul/Tema HKI	Jenis	Nomor P/ID

**VIII PENGALAMAN MERUMUSKAN KEBIJAKAN PUBLIK/REKAYASA SOSIAL LAINNYA**

No.	Tahun	Judul/Tema/Jenis Rekayasa Sosial Lainnya yang Telah Diterapkan	Tempat Penerapan	Respons Masyarakat

Semua data yang saya isikan dan tercantum dalam biodata ini adalah benar dan dapat dipertanggungjawabkan secara hukum. Apabila di kemudian hari ternyata dijumpai ketidaksesuaian dengan kenyataan, saya sanggup menerima risikonya.

Demikian biodata ini saya buat dengan sebenarnya untuk memenuhi salah satu persyaratan dalam pengajuan hibah Penelitian Kompetensi.

Bandung, 7 Desember 2012

Peneliti

(Dr. Ing., Yuliadi Erdani, M.Sc.)

LAMPIRAN 5: PUBLIKASI

# Publikasi Jurnal Internasional

Makalah dipublikasikan pada:

## **IJACSA**

*(International Journal of Advanced Computer Science  
and Applications)*

Vol. 3 Issue 9, 2012  
ISSN 2156-5570 (Online)  
ISSN 2158-107X (Print)  
[www.ijacsa.thesai.org](http://www.ijacsa.thesai.org)

# Developing Backward Chaining Algorithm of Inference Engine in Ternary Grid Expert System

Yuliadi Erdani  
Politeknik Manufaktur Bandung  
Bandung, Indonesia

**Abstract**— The inference engine is one of main components of expert system that influences the performance of expert system. The task of inference engine is to give answers and reasons to users by inference the knowledge of expert system. Since the idea of ternary grid issued in 2004, there is only several developed method, technique or engine working on ternary grid knowledge model. The in 2010 developed inference engine is less efficient because it works based on iterative process. The in 2011 developed inference engine works statically and quite expensive to compute. In order to improve the previous inference methods, a new inference engine has been developed. It works based on backward chaining process in ternary grid expert system.

This paper describes the development of inference engine of expert system that can work in ternary grid knowledge model. The strategy to inference knowledge uses backward chaining with recursive process. The design result is implemented in the form of software. The result of experiment shows that the inference process works properly, dynamically and more efficient to compute in comparison to the previous developed methods.

**Keywords**— *expert systems; ternary grid; inference engine; backward chaining.*

## I. INTRODUCTION

There is no official definition for the term of expert system but there are some descriptions for it created by people working in the field of expert system. With the term of expert system we refer to a computer system or a program, into which several procedures of artificial intelligence are integrated. Expert system can be understood as vehicles for Artificial Intelligence (AI) techniques.

Expert system is also applied artificial intelligence. Expert systems are programs for storing and processing knowledge of a special area, that why they are able to answer to questions and solve problems, with which experts normally deal [6]. In the current situation, expert system is an intelligent computer program that uses knowledge and inference procedures to solve problems that are difficult enough to require significant human expertise for their solution.

The expert knowledge must be obtained from specialist or other sources of expertise, such as texts, journal, articles, and database [8]. This type of knowledge usually requires much training and experience in some specialized field such as medicine, geology, system configuration, or engineering design. Once a sufficient body of expert knowledge has been acquired, it must be encoded in some form, loaded into a

knowledge base, then tested, and refined continually throughout the life of the system.

Some task that can be performed by expert system are difficult tasks to be specified, the task that may have incomplete or uncertain data, there may not always be an optimum solution, the task cannot be solved in a step-by-step manner, and solutions are often obtained by using accumulated experience [1]. An example of applied expert system is web-based consultation system [1]. Benefit of expert systems is the ability to preserve valuable knowledge which would otherwise be lost when an expert system is no longer available. Expert system also can allow an expert to concentrate on more difficult aspect of the task. It can enforce consistency, and they can perform dangerous tasks which would otherwise be carried out by humans.

One of known and very popular expert system type is production rule. Production rule are simple but powerful forms of knowledge representation providing the flexibility of combining declarative and procedural representation for using them in a unified form. The term production rule came from production system which is developed by A production system is a model of cognitive processing, consisting of a collection of rules (called production rules, or just productions). Each rule has two parts: a condition part and an action (conclusion) part. The meaning of the rule is that when the condition holds true, then the action is taken. A typical production rule is given below:

**IF there is a flame THEN there is fire**

The statement of the rule above means that fire is caused by a flame. If anything happens with a flame, it will lead to fire production. It is the idea of production system. The production system or production rule provides appropriate structures for performing and describing search process. A production system has four basic components as enumerated below [9]: A set of rules following the classical IF-THEN construct. If the conditions on the left-hand side are satisfied, the rule is fired, resulting in the performance of action on the right-hand side of the rule, A database of current facts established during the process of inference, A control strategy which specifies the order in which the rule are selected for matching of antecedents by comparing the facts in the database. It also specifies how to resolve conflicts in selection of rule or selection of facts, and a rule firing module. An expert system that implements production rule is known as rule-based expert system.

Building or construction of rules can easily be done in most rule-based expert system. Knowledge expert or knowledge engineer does not have to do any work specifying rules and how they are linked to each other. Sometime the knowledge expert or knowledge engineer can reference rules or facts that have not yet been created. It seems to be a simple and an instant work. The problem due to the performance of the knowledge will not occur until the number of rules is getting higher. Some problem may appear in the form of inconsistent rules, unreachable rules, redundant rule and closed rule chain of rules.

The mentioned problems above have been solved with so called Ternary Grid [1][4][5]. Since the ternary grid can solve some problem concerning knowledge bottleneck, there is no any developed inference method, technique or machine working on ternary grid knowledge model. As consequence of it, all ternary grid knowledge must be converted into production system format, so that the knowledge can be processed by rule-based inference machine to deliver solution. The inference engine of an expert system interprets and evaluates the facts in the knowledge base in order to provide an answer. By the numerous methods of problem solution, which can be implemented in a rule interpreter, only the representatives of the concatenation strategies are to be treated here: forward chaining and backward chaining

The developed inference machine of expert system can work in ternary grid knowledge model. The strategy to find solution previously uses forward chaining with iterative approach [12]. As another alternative solution, the inference machine can be implemented by using backward chaining method. The emphasis of this paper is to describe the backward chaining method that is implemented in the inference engine of ternary grid expert system. The backward chaining method should bring more benefit than developed forward chaining method, e.g. dynamic answers of inference engine, efficient computation effort, etc.

II. METHODS

Before talking the inference engine, we must first regard the knowledge representation. The Ternary Grid represents the production rule in the following structure (Fig. 1):

	F <sub>j</sub>
R <sub>i</sub>	0/1/2

Figure 1. Ternary Grid basic structure

- R<sub>i</sub>: Rule i (i is the number of rule)
- F<sub>j</sub>: Fact j or logical term (j is the number of fact)
- $i = \{1, 2, 3, \dots, I\}$
- $j = \{1, 2, 3, \dots, J\}$
- $J > I + 1$
- The Value of every grid box is 0, 1 or 2
- 0 = unused, is represented by empty grid box.

1 = Fact F<sub>m</sub> belongs to the condition part of rule R<sub>n</sub> (LHS= Left Hand Side).

2 = Fact F<sub>m</sub> is part of the conclusion part of R<sub>n</sub> (RHS = Right-Hand Side).

In the beginning of the development, the Ternary Grid was only used for knowledge acquisition system. The basic feature of the system architecture is to organize the independent and sequential obtaining process of the factual knowledge and the elicitation process of judgmental knowledge using Ternary Grid. The overall systems architecture is presented in terms of collection of functions providing effective acquisition, processing, transferring and flexible transformation of knowledge. This section gives an overview of the system that shows the design approach of the system and the concept of acquisition process.

The Ternary Grid acquisition system has task to organize the knowledge base, to obtain the factual knowledge, to elicit the judgmental knowledge, and to transfer the knowledge into knowledge base. The system was able to improve the performance of expert system knowledge [5]. Meanwhile the Ternary Grid has been used for other part related the expert system, such as knowledge representation, knowledge based system, inference engine, etc.

Even the Ternary Grid has been applied in an inference engine, but the approach of existing inference engine uses only forward chaining method. The backward chaining method has not been used in any inference engine. The developed backward chaining method will be implemented in inference engine of expert system based on Ternary Grid. Inference engine of expert system is computer program that answers questions from user. It processes all information from the knowledge base by firing rules and facts [9].

Backward chaining is a strategy of inference process which is the opposite of forward chaining. The strategy of backward chaining is started from a goal and ended with a fact that leads to the goal. Backward chaining method is also called as goal driven strategy of inference engine. In other literature, the backward chaining is a chaining process that begins with the last element in the chain and proceeds to the first element. This is often a very effective way of developing complex sequences of behavior

There are two search algorithms, which are normally used by backward chaining method, i.e. depth-first and breath-first search algorithm. Both algorithms search data in a tree structure. Depth-first search algorithm searches a data in a tree as deep as possible before backtracking. Breath-first search algorithm searches a data in neighbor nodes before it moves deeper to the bottom of the tree. Using depth-first search algorithm, the process of backward chaining can be illustrated as follows:

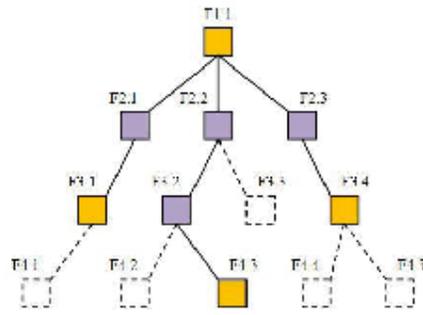


Figure 2. Backward chaining illustration

Figure 3 shows the illustration of backward chaining process of inference engine of expert system. The known facts are F1.1, F2.1, F3.2 and F3.4 in the beginning of process. The inference process begins from fact F1.1. That fact F1.1 is called as goal. The inference process moves then backward to other facts behind goal. It is so called condition part of rule. The inference engine tries to applied fact F2.1, F2.2 and F2.3, etc. The developed backward chaining uses depth-first search algorithm. The following steps describe the inference process of expert system using developed backward chaining algorithm as follows:

- Goal: F1.1
- Fetch: F2.1 → unknown
- Fetch: F3.1 → known
- Rule (F3.1, F2.1) → fired
- Rule (F2.1, F1.1) → fired
- Fetch: F4.1 → unknown
- Fetch: F2.2 → unknown
- Fetch: F3.2 → unknown
- Fetch: F4.2 → unknown
- Fetch: F4.3 → known
- Rule (F4.3, F3.2) → fired
- Rule (F3.2, F2.2) → fired
- Rule (F2.2, F1.1) → fired
- ...
- Etc.

The inference process continues until all possible facts have been asked (tested) to be fired. The developed algorithm searches all data deeper into the bottom of tree structure in the knowledge base of expert system. The designed and implemented backward chaining algorithm is explained in the following algorithm:

```

Backward Chaining (...)
While: Rule Number == Existing Rules ...()
  Rule is not applicable Do
  IF Rule applicable: True
    Incrementing Number of Working Node
  IF Not End of Rule: True
    Incrementing Number of Active Node
    Listing Active Node
    Technical Chaining (Real Node, ...)
  Else
    IF Number of Active Node = 0 Then
      First Element — List of Active Node
      Decrementing Number of Active Node
      Technical Chaining (Search Node, ...)
  
```

The process continues as far as the number of rules is less than the number of existing rules and there is rule that is not applicable. If a rule is applicable (fired) then the program search the next facts that lead to applicable rule until there is no fact found anymore. If a rule is not applicable (fired) then the program search other possible rule in other paths. The process continues until all possible facts have been tested or fetched.

III. RESULTS

The same data as [12] [13] is used in this experiment.

```

Rule 1: IF unusual beam > 500 AND whose shaft is finished - THEN internal result passed.
Rule 2: IF received result previous AND gas analysis score greater than or equal 60% AND no repetitive test previous - THEN final project passed.
Rule 3: IF final project passes AND others evaluation on time - THEN graduation on time.
Rule 4: IF final project passes & AND others evaluation on time - THEN graduation prepared.
  
```

According to Ternary Grid acquisition technique [5], the mentioned rules are inputted into ternary grid knowledge base as it is shown in figure 4. Using the developed concept, the rule-based format must not be converted into ternary grid. The inference process of the expert system in ternary grid uses backward chaining with recursive approach. All fact inputs are stored in set of facts Fk. The inference engine searches all rules that are possible to be executed and stores them in set of rules Rx:

$$R_x = \{p | p \rightarrow q, p \in F_k, p \in F, F_k \subset F\} \quad (5)$$

The inference engine determines then rules that are able to be applied and stores in the following set of rule Ryn.

$$R_{yn} \subset R_x \quad (6)$$

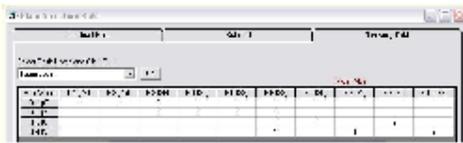


Figure 3. Given facts and rules in ternary grid

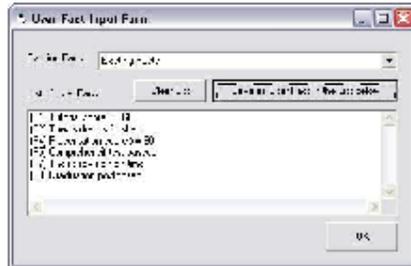


Figure 4. Known facts

The application does then the inference task by processing all facts that are given before. The result of inference process can be shown in figure 6.

Inconsistent rules can be detected and eliminated using the following processes:

- Find rows, in which value 3 appears:

$$B = \{i \mid a_{ij} = 3\} \quad (7)$$

- Remove row duplication

$$C = \bigcup_{i \in B} b \quad (8)$$

The result of inference process shows the effectiveness of the developed algorithm. In comparison to the method of [7] [12] and [13], the developed inference method can work directly in ternary grid without having to be converted to rule-based format. In comparison to [12] and [13], the developed method work more dynamic and efficient to compute. The implemented recursive approach in inference process reduced the number of required iteration. The result of inference process can also show other facts that weren't known before. These all facts could lead to give more conclusions that will bring more information to expert system.

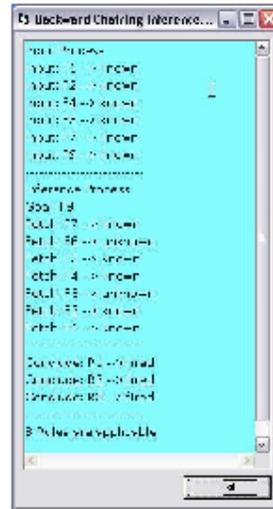


Figure 5. Result of inference process using backward chaining algorithm

The following data are taken from several conducted experiments

TABLE I. EXPERIMENT DATA		
Number of rule	Number of fact	Number of Iteration
10	35	62
20	70	134
30	100	295
50	180	673

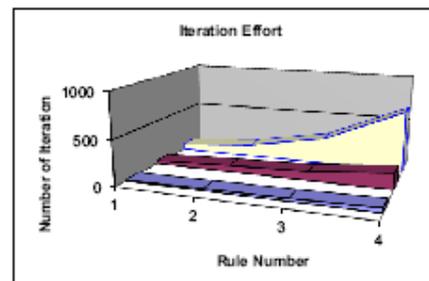


Figure 6. Recursion effort

Figure 6 show the effort of recursion that is influenced by increasing the number of facts and rules.

IV. CONCLUSION

The developed inference engine using backward chaining method in ternary grid works properly. It can determine all applied rules that lead to the goal fact. In comparison to the previous work using iterative approach and forward chaining algorithm, the developed method works more dynamic and more efficient to compute. The inference process could also detect and lead to other rules that previously unknown and brought new solutions. Referring to some literatures concerned expert systems; the developed method is novel and will give contribution in developing inference method of expert systems.

REFERENCES

[1] David J.C. McKay. Information Theory, Inference, and Learning Algorithms. Cambridge University Press. 2003.

[2] Erdani, Y., Hunger, A., Werner, Stefan, Mertens, S. Web-Based Consultation System with Expert System, IASTED CST 2003 – International conference (International Association of Science and Technology for Development – Computer Science and Technology), May 19-21, 2003, Cancun, Mexico, 2003, ISBN – 0-88986-349-0, page 61-64

[3] Erdani, Y., Hunger, A., Werner, S., Mertens, S. Ternary Grid as a Potentially New Technique for Knowledge Elicitation/Acquisition, 2nd IEEE Conference on Intelligent System, vol 1: pp. 312-315. ISBN 0-7803-8278-1, Vama - Bulgaria, June 22-24, 2004. This paper has been accepted also by SEKE 04 (Software Engineering and Knowledge Engineering) conference in Banff, Canada.

[4] Erdani, Y., Hunger, A., Werner, S., Improving the Knowledge Performance using Ternary Grid Knowledge Acquisition and Model, WSEAS Transactions (Journals) on Information Science and Application, Issue 2, Volume 2, February 2005. ISSN 1790-0832

[5] Erdani, Yuliadi, Acquisition of Human Expert Knowledge for Rule-based Knowledge-based Systems using Ternary Grid, Verlag – Dissertation.de, Berlin, 2005, ISBN 3-89825-000-8

[6] Horn, Christian; Kerner, Immo O. : Lehr- und Übungsbuch Informatik, Band 3. Fachbuchverlag Leipzig, im Carl Hanser Verlag, München Wien, 1997. ISBN 3-446-18699-9

[7] Hunger, A., Werner, S., CONGA: A Course Online/Offline Information and Guidance System to support an International Degree Course, Proceeding of ICCE 99 (Chiba-Japan, 1999, ISBN 1 58603 027 2, Page 577-583)

[8] Joseph C. Giarratano, Gary Riley. Expert Systems, Principles and Programming 2005, ISBN 0-534-38447-1

[9] Krishnamoorthy, C. S., Rajeev, S., Artificial Intelligence and Expert Systems for Engineer, CRC Press, Boca Raton – Florid, 1996, ISBN 0-8493-9125-3

[10] Lunze, Prof. Dr.-Ing. Jan. Künstliche Intelligenz für Ingenieure, Band 1: Methodische Grundlagen und Softwaretechnologie, R. Oldenbourg Verlag, München Wien, 1994, ISBN 3-486-22287-2

[11] Sascha Mertens, Marius Rosa, Yuliadi Erdani. An intelligent dialogue for online rule based expert systems, Proc. of the 2004 International Conference on Intelligent User Interfaces, January 13-16, 2004, Funchal, Madeira, Portugal. ACM 2004, ISBN 1-58113-815-6

[12] Yuliadi Erdani, Pengembangan Metode Inferensi untuk Sistem Pakar berbasis Ternary Grid, Jurnal P&PT Vol. VIII, No. 2, Desember 2010, ISSN 0854-5766

[13] Yuliadi Erdani, Developing Recursive Forward Chaining Method in Ternary Grid Expert Systems, IJCSNS (International Journal of Computer Science and Network Security), Vol. 11 No. 8, August 2011, ISSN 1738-7906

AUTHORS PROFILE



Dr. Ing. Yuliadi Erdani, M.Sc., Dipl. EL. Ing., HTL received the B.S. degree in Electrical Engineering from the Ingenieurschule Burgdorf (ISB) Switzerland in 1995 and the M.S. degree in Computer Science and Communication Engineering (CSCE) from the University of Duisburg, Germany in 2002. He received PhD degree (Dr.-Ing.) in Informationstechnik from the same university i.e. University of Duisburg-Essen, Germany in 2005. He did a lot of work with expert systems. He has been continuing his research work in the area of expert systems from 2006 until now and every year he always got research fund/grant. He works now as lecture in a state Polytechnic (university of applied science) in Bandung, Indonesia.

LAPORAN KEUANGAN



**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN  
POLITEKNIK MANUFAKTUR NEGERI BANDUNG**

JALAN KANAYAKAN 21, DAGO - BANDUNG 40135; TELEPON (022) 2500241 FAKSIMILE: (022) 2502649  
homepage: <http://www.polman-bandung.ac.id> e-mail: [sekretariat@polman-bandung.ac.id](mailto:sekretariat@polman-bandung.ac.id)

**BUKU KAS UMUM**

Program : Hibah Kompetensi Tahun ke 3  
Ketua Peneliti : Dr. Ing. Yuliadi Erdani, M.Sc.  
Judul Penelitian : Pengembangan Metode Inferensi untuk Sistem Pakar berbasis Ternary Grid

Penerimaan				Pengeluaran			
Tanggal	Uraian	No. Bukti	Jumlah	Tanggal	Uraian	No. Bukti	Jumlah
26-Jun-12	Dana Penelitian 70%	Kontrak	59.500.000				
				02-Jul-12	Pembuatan Add. Fact & Rule Database	Kuitansi	10.000.000
				02-Jul-12	Pembuatan Software Simulator Inferensi Backward Chaining	Kuitansi	20.000.000
				02-Jul-12	Honor Peneliti Tahap I	Kuitansi	5.100.000
				02-Jul-12	PPH 21 Final 5%	SSP	2.975.000
				05-Agust-12	Publikasi Jurnal Internasional IJACSA	Kuitansi	3.045.000
				06-Agust-12	Pembuatan Software Integrator Interface	Kuitansi	15.000.000
				03-Sep-12	Dokumentasi, Administrasi, Operasional, Perjalanan, dll.	Kuitansi	3.380.000
26-Nop-12	Dana Penelitian 30%	Kontrak	25.500.000				
				30-Nop-12	PPH 21 Final 5%	SSP	1.275.000
				05-Des-12	Honor Peneliti Tahap II dan III	Kuitansi	11.900.000
				06-Des-12	Dokumentasi, Administrasi, Operasional, Perjalanan, dll.	Kuitansi	12.325.000
<b>Jumlah</b>			<b>85.000.000</b>				<b>85.000.000</b>

Mengetahui  
Kepala UP3M/LPPM



Dr. Ing. Yuliadi Erdani, M.Sc.  
NIP. 196807021997021001

Bandung, 7 Desember 2012

Ketua Peneliti

Dr. Ing. Yuliadi Erdani, M.Sc.  
NIP. 196807021997021001