

Studi Komparasi Terhadap Kapabilitas Generalisasi dari Jaringan Saraf Tiruan Berbasis *Incremental Projection Learning*

Hendri Murfi*, Benyamin Kusumoputro**

*Jurusan Matematika FMIPA - Universitas Indonesia, Depok 16424
e-mail: hendri@makara.cso.ui.ac.id

**Fakultas Ilmu Komputer - Universitas Indonesia, Depok 16424
e-mail: kusumo@cs.ui.ac.id

Abstrak

Salah satu hal yang penting dari suatu metode pembelajaran pada jaringan saraf tiruan adalah kapabilitas generalisasi. Yaitu kemampuan untuk memberikan hasil yang akurat terhadap data yang tidak diajarkan pada tahap pembelajaran. Salah satu metode pembelajaran yang memberikan jaminan secara teori diperolehnya kapabilitas generalisasi yang optimal adalah *projection learning*. Pada tulisan ini kami akan melakukan evaluasi eksperimental terhadap kapabilitas generalisasi dari jaringan saraf tiruan berbasis *projection learning* yang bersifat *incremental*, yang disebut *projection generalizing neural networks*, untuk memecahkan masalah aproksimasi fungsi. Kemudian melakukan studi komparasi dengan jaringan saraf tiruan yang sudah umum digunakan, yaitu *back propagation networks* dan *radial basis functions networks*. Berdasarkan hasil uji coba komputasi yang kami lakukan, *projection generalizing neural networks* tidak selalu memberikan kapabilitas generalisasi yang lebih baik. *projection generalizing neural networks* memberikan kapabilitas generalisasi yang lebih baik ketika jumlah data pembelajaran cukup kecil atau variansi *noise* dari data pembelajaran cukup besar. Selain dari dua kondisi tersebut, *projection generalizing neural networks* tidak selalu memberikan kapabilitas generalisasi yang lebih baik. Bahkan, untuk kondisi dimana jumlah data pembelajaran cukup besar dan variansi *noise* cukup kecil, *projection generalizing neural networks* memberikan kapabilitas generalisasi yang lebih buruk dari *back propagation networks*.

Kata kunci: metode pembelajaran dengan pengarah (supervised learning), *incremental projection learning*, kapabilitas generalisasi, jaringan saraf tiruan, masalah aproksimasi fungsi.

Abstract

One of the essences of supervised learning in neural network is generalization capability. It is an ability to give an accurate result for data that are not learned in learning process. One of supervised learning method that theoretically guarantees the optimal generalization capability is *incremental projection learning*. This paper will describe an experimental evaluation of generalization capability of the *incremental projection learning* in neural networks, called *projection generalizing neural networks*, for solving function approximation problem. Then, Make comparison with other general used neural networks, i.e. *back propagation networks* and *radial basis function networks*. Base on our experiment, *projection generalizing neural networks* doesn't always give better generalization capability than the two other neural networks. It gives better generalization capability when the number of learning data is small enough or the noise variance of learning data is large enough. Otherwise, it does not always give better generalization capability. Even though, In case the number of learning data is big enough and the noise variance of learning data is small enough, *projection generalizing neural networks* gives worse generalization capability than *back propagation networks*.

Keywords: supervised learning, *incremental projection learning*, generalization capability, artificial neural networks, function approximation problem

Pendahuluan

Salah satu hal yang penting dari suatu metode pembelajaran dengan pengarah (*supervised learning*) pada jaringan saraf tiruan adalah kapabilitas generalisasi (*generalization capa-*

bility). Yaitu Kemampuan untuk untuk memberikan hasil yang akurat terhadap data yang tidak diajarkan pada tahap pembelajaran. Kapabilitas generalisasi ini dapat diukur berdasarkan nilai *generalization error* yang dihasilkan oleh metode tersebut. Semakin kecil nilai *generalization error* semakin baik kapabilitas generalisasinya.

Catatan : Diskusi untuk makalah ini diterima sebelum tanggal 1 Januari 2002. Diskusi yang layak muat akan diterbitkan pada Jurnal Teknik Elektro volume 2 nomor 1 Maret 2002

Generalization error = $E\|f_m - f\|$, dimana E adalah fungsi ekspektasi, f_m adalah fungsi aproksimasi dan f adalah fungsi target. Dikembangkan oleh Takemura menjadi $\|Ef_m - f\|^2 + E\|f_m - Ef_m\|^2$, dimana suku pertama disebut bias dan suku kedua disebut variansi.

Salah satu metode pembelajaran yang dibuat dengan meminimumkan *generalization error* secara langsung adalah *projection learning* yang dikembangkan oleh Ogawa [5]. Ogawa merumuskan metode pembelajaran tersebut sebagai suatu masalah inverse (*inverse problem*) dengan pendekatan analisa fungsional. Pada metode ini, fungsi target pembelajaran dan fungsi hasil pembelajaran diasumsikan berada pada ruang *reproducing kernel Hilbert*. Dalam mendapatkan fungsi hasil pembelajaran, metode ini akan meminimalkan variansi dan juga mereduksi bias dari fungsi hasil pembelajaran tersebut terhadap fungsi target. Bias direduksi dengan cara menempatkan fungsi hasil pembelajaran sebagai proyeksi orthogonal dari fungsi target. Dengan pendekatan seperti ini maka metode *projection learning* akan meminimumkan bias dan juga variansi dari *generalization error*. Kondisi inilah yang memberikan jaminan secara teori bahwa metode *projection learning* akan menghasilkan kapabilitas generalisasi yang optimal. Jaringan saraf tiruan yang merepresentasikan fungsi hasil pembelajaran dari *projection learning* ini disebut *projection generalizing neural networks* (PGNN)

Pada jaringan saraf tiruan adalah sesuatu yang sering terjadi untuk meningkatkan kapabilitas generalisasi setelah proses pembelajaran selesai. Salah satu cara untuk melakukan hal tersebut adalah menambahkan data pembelajaran ke jaringan saraf tiruan tersebut. Ditinjau dari sudut pandang pembelajaran pada diri manusia maka cara seperti ini adalah alami. Yaitu membangun pengetahuan berdasarkan pengetahuan sebelumnya. Metode pembelajaran seperti ini disebut juga dengan istilah *incremental learning* [7]. *Projection learning* juga sudah dikembangkan untuk pembelajaran yang bersifat *incremental* yang dikenal dengan istilah *incremental projection learning* (IPL). Vijayakumar dan Ogawa mengembangkan IPL dengan meniadakan *noise* pada data pembelajarannya [10]. Kemudian dilanjutkan oleh Sugiyama dan Ogawa untuk data pembelajaran yang mengandung *noise* [7][8]. Sugiyama dan Ogawa

juga menunjukkan bahwa hasil yang didapat dari IPL memiliki kapabilitas generalisasi yang sama dengan yang dihasilkan oleh *batch projection learning* (BPL). Sementara umumnya metode pembelajaran yang bersifat *incremental* memberikan kapabilitas generalisasi yang lebih buruk dari metode pembelajaran yang bersifat *batch*. Hasil ini sangat penting, karena disamping menjamin bahwa IPL memberikan kapabilitas generalisasi yang optimal, hasil ini menunjukkan juga bahwa peluang untuk meningkatkan kapabilitas generalisasi dari IPL masih mungkin apabila digabungkan dengan pembelajaran yang aktif (*active learning*).

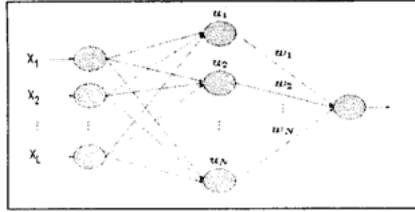
Pada tulisan ini, kami akan melakukan studi komparasi kapabilitas generalisasi dari PGNN berbasis IPL terhadap jaringan saraf tiruan yang sudah umum digunakan, yaitu *back propagation* (BP) *networks* [3][2] dan *radial basis function* (RBF) *networks* [1][2]. Uji coba komputasi dilakukan dengan menggunakan perangkat lunak Matlab.

Tulisan ini diorganisasikan sebagai berikut: bagian 2 akan menjelaskan tentang arsitektur dan algoritma dari PGNN berbasis IPL. Pada bagian 3 akan diberikan hasil uji coba komputasi dan simulasi komputer terhadap penggunaan PGNN berbasis IPL dalam memecahkan suatu masalah ujicoba buatan. Studi komparasi dengan BP dan RBF juga akan diberikan pada bagian ini. Tulisan ini akan ditutup dengan suatu kesimpulan berdasarkan hasil studi perbandingan.

Arsitektur Dan Algoritma

PGNN adalah suatu jaringan saraf tiruan *feedforward* yang terdiri dari tiga lapisan (*three-layer feedforward neural networks*), yaitu: lapisan input, lapisan tersembunyi dan lapisan keluaran (Gambar 2.1). Lapisan output hanya terdiri-dari satu *neuron* sementara pada lapisan tersembunyi bersifat dinamis. Dimulai dengan tidak ada *neuron* pada awal pembelajaran, selanjutnya akan bertambah satu demi satu selama proses pembelajaran. Fungsi aktivasi hanya ada pada lapisan tersembunyi dan menggunakan fungsi *reproducing kernel*. Pada arsitektur PGNN ini tidak ada bobot antara lapisan input dan lapisan tersembunyi. Penentuan jumlah *neuron* pada lapisan tersembunyi dan nilai bobot dilakukan berdasarkan algoritma IPL yang akan dijelaskan pada bagian selanjutnya.

**Studi Komparasi Terhadap Kapabilitas Generalisasi dari Jaringan Saraf Tiruan
Berbasis Incremental Projection Learning**
{ Hendri Murfi et.al. }



Gambar 1 Arsitektur Jaringan

Setelah tahap pembelajaran selesai, maka fungsi hasil pembelajaran-nya adalah kombinasi linear dari fungsi basis dengan koefisiennya adalah nilai bobot yang bersesuaian. Selanjutnya nilai aproksimasi dari suatu input (misal: x) dapat dirumuskan sebagai berikut:

$$f_m(x) = \sum_{i=1}^N w_i u_i(x) \quad (1)$$

PGNN adalah suatu jaringan saraf tiruan yang merepresentasikan fungsi hasil pembelajaran yang dihasilkan oleh algoritma IPL. Dalam pengembangannya, Metode pembelajaran ini dibagi dalam dua tahap, yaitu membentuk fungsi aproksimasi pada tahap pertama. Selanjutnya, membangun jaringan saraf tiruan yang merepresentasikan fungsi aproksimasi tersebut pada tahap kedua. Pada tulisan ini kami menggunakan IPL yang dikembangkan oleh Sugiyama dan Ogawa [7][8]. Proposisi berikut ini adalah salah satu IPL yang dikembangkan untuk kasus dimana noise dari data pembelajaran berdistribusi normal $N(0, \sigma)$ dan variance σ positif.

Proposisi 1 [7][8] Jika matrik korelasi *noise* adalah positif definit dan diagonal maka fungsi hasil pembelajaran f_{m+1} diperoleh dari fungsi hasil pembelajaran sebelumnya f_m sebagai berikut

$$f_{m+1} = \begin{cases} f_m + \frac{\beta_{m+1} V_m^* \psi_{m+1}}{v_2^{(m+1)}} & \text{jika } \psi_{m+1} \in R(A_m^*) \\ f_m + \frac{\beta_{m+1} \tilde{\psi}_{m+1}}{v_1^{(m+1)}} & \text{jika } \psi_{m+1} \notin R(A_m^*) \end{cases} \quad (2)$$

dimana

$$V_m = A_m^* Q_m^{-1} A_m \quad (3)$$

$$\beta_{m+1} = y_{m+1} - f_m(x_{m+1}) \quad (4)$$

$$\tilde{\psi}_{m+1} = P_{N(A_m)} \psi_{m+1} \quad (5)$$

$$v_1^{(m+1)} = \tilde{\psi}_{m+1}^* (x_{m+1}) \quad (6)$$

$$v_2^{(m+1)} = \sigma_{m+1} + \langle V_m^* \psi_{m+1}, \psi_{m+1} \rangle \quad (7)$$

dengan ψ_{m+1} adalah fungsi sampling dari data pembelajaran ke- $m+1$, A_m adalah operator sampling, $R(A_m^*)$ adalah *range* dari adjoin A_m atau dalam hal ini adalah ruang aproksimasi, $P_{N(A_m)} \psi$ adalah proyeksi orthogonal dari ψ pada ruang *null* A_m , Q_m adalah matrik korelasi *noise* dan σ adalah variansi *noise*.

Langkah berikutnya adalah membangun jaringan saraf tiruan yang merepresentasikan fungsi hasil pembelajaran yang dihasilkan oleh algoritma IPL. Algoritma yang dibuat untuk membangun jaringan saraf tiruan berdasarkan fungsi hasil pembelajaran yang dihasilkan oleh algoritma IPL ini dikenal dengan nama *incremental projection learning in neural network* (IPLNN). Ada beberapa algoritma IPLNN yang sudah dikembangkan oleh Sugiyama dan Ogawa. Pada simulasi komputer yang akan kami lakukan pada bagian 4, kami menggunakan algoritma IPLNN4 yang merepresentasikan fungsi hasil pembelajaran (2) (Gambar 2) [6][9].

Pada algoritma IPLNN4 ini proses rekonstruksi jaringan dibagi menjadi dua kategori berdasarkan fungsi sampling dari data pembelajaran (ψ_i) terhadap ruang aproksimasi ($R(A_m^*)$). Jika $\psi_{m+1} \in R(A_m^*)$ maka satu *neuron* dengan fungsi aktifasi-nya adalah fungsi *reproducing kernel* $K(x, x_{m+1})$ ditambahkan ke lapisan tersembunyi. Selanjutnya bobotnya ditambahkan sementara bobot-bobot lainnya diperbarui. Sebaliknya, Jika $\psi_{m+1} \notin R(A_m^*)$ maka tidak ada penambahan *neuron*. Yang ada hanyalah memperbarui bobot-bobot yang ada.

Kondisi $\psi_{m+1} \in R(A_m^*)$ artinya bahwa ψ_{m+1} bebas linear terhadap $\{\psi_j\}_{j=1}^m$ dan ruang aproksimasi $R(A_{m+1}^*)$ menjadi lebih luas dari $R(A_m^*)$. Sebaliknya, $\psi_{m+1} \notin R(A_m^*)$ artinya bahwa ψ_{m+1} tidak bebas linear $\{\psi_j\}_{j=1}^m$ dan ruang aproksimasi $R(A_{m+1}^*)$ sama dengan $R(A_m^*)$. Kondisi $\psi_{m+1} \in R(A_m^*)$ dapat diperiksa karena $\psi_{m+1} \in R(A_m^*)$ jika dan hanya jika

$$P_{N(A_m)} \psi_{m+1} = \tilde{\psi}_{m+1} \neq 0 \quad (8)$$

Dalam implementasinya, Sugiyama dan Ogawa menggunakan kriteria berikut:

Jika $\|\tilde{\psi}_{m+1}\|^2 = v_1 > \varepsilon$ maka $\psi_{m+1} \in R(A_m^*)$ (9)

dimana ε adalah suatu konstanta yang kecil, misal $\varepsilon = 10^{-4}$.

Simulasi dan Analisa Hasil

Pada bagian ini akan diberikan hasil uji coba komputasi dan simulasi terhadap kapabilitas generalisasi dari PGNN berbasis IPL. Pada simulasi ini, kami menggunakan fungsi (10) sebagai masalah pengujian dengan domain berada pada interval $[-\Pi, \Pi]$. *Generalization error* dihitung berdasarkan persamaan (11).

$$f = 2x - 14e^{-3(x-2.5)^2} - 5e^{-6(x-0.5)^2} + 3e^{-3x^2} + 12e^{-1(x+2.5)^2} \quad (10)$$

$$Gen. err = \frac{1}{126} \sum_{n=0}^{125} [f(-\pi + 0.05n) - f_n(-\pi + 0.05n)]^2 \quad (11)$$

Selanjutnya akan dilakukan studi komparasi dengan Jaringan *on-line Back Propagation* (BP) berbasis *gradient descend learning with momentum* dan jaringan *Radial Basis Function* (RBF) berbasis *orthogonal least squares learning*. Karakteristik dari masing-masing jaringan saraf tiruan tersebut adalah sebagai berikut:

- PGNN. Ruang *reproducing kernel Hilbert*-nya direntang oleh $\{1, \sin kx, \cos kx\}$ dengan k adalah orde dari basis. Sementara *inner product*-nya didefinisikan sebagai berikut:

$$\langle f, g \rangle = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x)g(x)dx \quad (12)$$

Dan *reproducing kernel* didefinisikan sebagai berikut:

$$K(x, x') = \begin{cases} 2k+1 & \text{jikax} = x' \\ \frac{\sin \frac{(2k+1)(x-x')}{2}}{2} & \text{jikax} \neq x' \end{cases} \quad (13)$$

```

input (xm+1, ym+1), σm+1
if m = 0 {
    generate first neuron in hidden layer
    u1 ← K(x, x1); w1 ← -y1 / ||ψ1||2;
    C1(1) ← σ1 / ||ψ1||2; D1(1) ← 1 / ||ψ1||2;
    N ← 1;
} else {
    [bN(m+1)]i ← ⟨ui, ψm+1⟩;
    [cN(m+1)]i ← ∑j=1N [CN(m)]j [bN(m+1)]j;
    [dN(m+1)]i ← ∑j=1N [DN(m)]j [bN(m+1)]j;
    βm+1 ← ym+1 - ∑i=1N wi [bN(m+1)]i;
    v1(m+1) ← ψm+1(xm+1) - ∑i=1N [cN(m+1)]i [bN(m+1)]i;
    v2(m+1) ← σm+1 + ∑i=1N [dN(m+1)]i [bN(m+1)]i;
    if v1(m+1) ≠ 0 {
        add the (N+1)-st neuron in hidden layer
        uN+1 ← K(x, xm+1), wN+1 ← βm+1 / v1(m+1);
        w(N) ← w(N) - βm+1 / v1(m+1) CN(m+1);
        CN+1(m+1) ← 1 / (N+1) CN(m) + (dN+1(m+1) ⊗ aN+1(m+1)) / v1(m+1);
        DN+1(m+1) ← ΓN+1 CN(m) ΓN+1 + (v2(m+1) (aN+1(m+1) ⊗ aN+1(m+1)) / v1(m+1)) - (ΓN+1 dN(m+1) ⊗ aN+1(m+1) + aN+1(m+1) ⊗ ΓN+1 dN(m+1)) / v1(m+1);
        N ← N + 1;
    } else {
        w(N) ← w(N) + βm+1 dN(m+1) / v2(m+1);
        CN(m+1) ← CN(m);
        DN(m+1) ← DN(m) - dN(m+1) ⊗ dN(m+1) / v2(m+1);
    }
}

```

Gambar 2 Pseudocode Algoritma IPGNN4

- BP. Arsitektur jaringan terdiri dari tiga lapisan. Jumlah neuron pada lapisan tersembunyi ditentukan secara manual dan bersifat statis selama proses pembelajaran. Fungsi aktivasi yang digunakan adalah fungsi *tag-sigmoid* pada lapisan tersembunyi dan fungsi linear pada lapisan output. Algoritma akan berhenti jika *epoch* sudah

**Studi Komparasi Terhadap Kapabilitas Generalisasi dari Jaringan Saraf Tiruan
Berbasis *Incremental Projection Learning***
[Hendri Murfi et.al.]

lebih besar dari 3000 atau nilai *mean square error* lebih kecil dari 10^{-5} atau nilai *generalization error* minimum sudah dicapai. Nilai bobot diperbarui secara *incremental (on-line)*.

- RBF. Arsitektur jaringan terdiri dari tiga lapisan. Jumlah neuron pada lapisan tersembunyi bersifat dinamis. Mulai dari tidak ada pada awal pembelajaran selanjutnya berkembang selama proses pembelajaran. Fungsi aktivasi yang digunakan adalah fungsi *radial basis* pada lapisan tersembunyi dan fungsi linear pada lapisan output. Algoritma akan berhenti jika jumlah maksimal *neuron* pada lapisan tersembunyi sudah tercapai atau nilai fungsi *sum square error* sudah lebih kecil dari 10^{-5} .

Perlu dicatat bahwa metode pembelajaran pada Jaringan BP dan RBF yang digunakan masih bersifat *batch*. Khusus untuk *orthogonal least squares learning* pada jaringan RBF sedang dikembangkan untuk pembelajaran yang bersifat *incremental*. Salah variasinya adalah algoritma *recursive orthogonal least squares* [4].

Simulasi nilai *generalization error* dikelompokkan berdasarkan jumlah dan nilai variansi *noise* dari data pembelajaran. Simulasi dilakukan dengan menggunakan perangkat lunak Matlab pada mesin PC IBM Think Pad dengan *processor* Pentium II berbasis windows 98.

Tabel 1 adalah hasil simulasi terhadap 40 data pembelajaran dengan *noise* berdistribusi $N(0,1)$. Dari tabel 4.1 dapat dilihat bahwa PGNN tidak selalu memberikan kapabilitas generalisasi yang terbaik. PGNN dapat memberikan kapabilitas generalisasi terbaik (kasus I), pertengahan (kasus II) atau bahkan terjelek (kasus III). Nilai kapabilitas generalisasi yang diberikan adalah nilai kapabilitas generalisasi dengan model yang optimal. Model disini adalah faktor-faktor yang mempengaruhi kapabilitas generalisasi dari masing-masing jaringan saraf tiruan tersebut. Yaitu nilai orde basis untuk PGNN, jumlah neuron pada lapisan tersembunyi untuk BP dan nilai faktor *spread* untuk RBF. Untuk BP terdapat faktor lain yang mempengaruhi, yaitu nilai inisialisasi untuk bobot. Sehingga, nilai *generalization error* yang diberikan adalah nilai rata-rata untuk beberapa nilai inisialisasi bobot tersebut.

Tabel 1. Tiga keadaan kapabilitas generalisasi dari PGNN

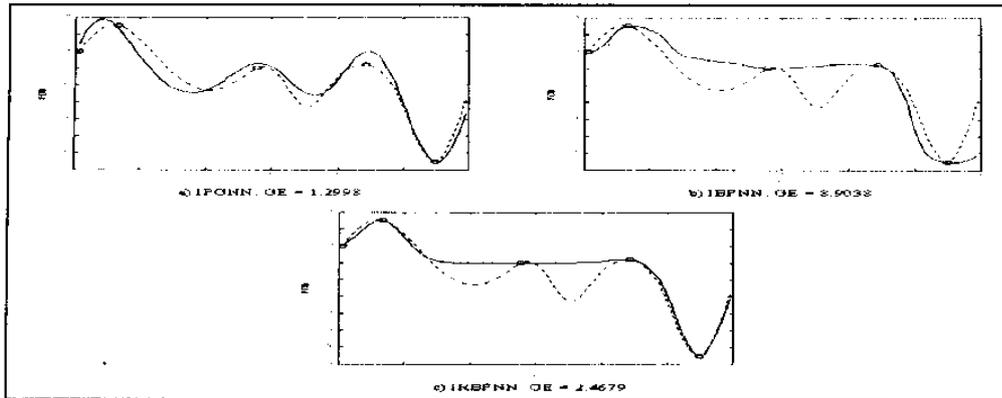
Kasus	Jumlah Data	Dist. Noise	IPGNN	IBPNN	IRBFNN
			GE	GE	GE
I	40	$N(0,1)$	0.3999	0.8573	1.1943
II	40	$N(0,1)$	1.1037	0.8904	3.5178
III	40	$N(0,1)$	0.9641	0.6966	0.7221

Selanjutnya akan disimulasikan *generalization error* berdasarkan jumlah data pembelajaran. Tabel 2 merupakan nilai *generalization error* untuk beberapa jumlah data pembelajaran. Dari Tabel 2 terlihat bahwa kapabilitas generalisasi PGNN cenderung lebih baik untuk data pembelajaran yang berukuran kecil, terutama jika dibandingkan dengan BP. Kita juga dapat melihat pada kasus II dan III bahwa nilai *generalization error* yang diberikan oleh PGNN untuk data berukuran kecil lebih baik walaupun pada data yang berukuran besar nilai *generalization error* yang diberikan lebih jelek. Simulasi grafis untuk kasus III dengan lima data pembelajaran diberikan pada Lampiran pada Gambar 3.

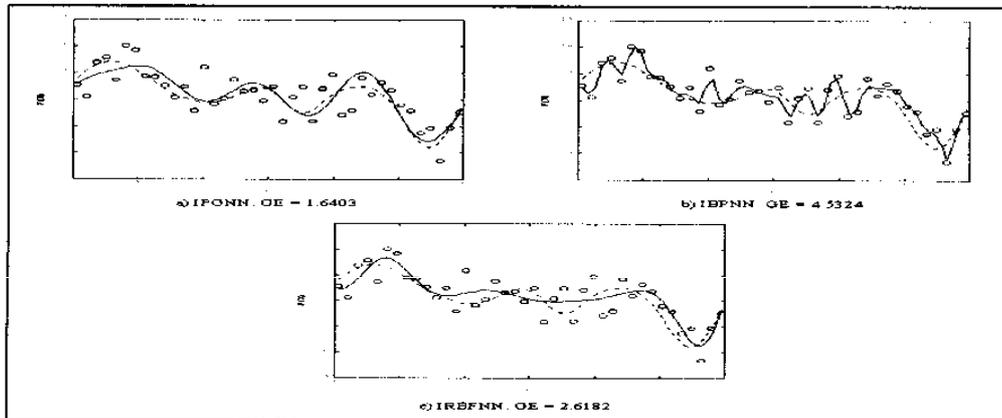
Tabel 2. Perbandingan kapabilitas generalisasi berdasarkan jumlah data

Kasus	Jumlah Data	Dist. Noise	IPGNN	IBPNN	IRBFNN
			GE	GE	GE
I	40	$N(0,1)$	0.3999	0.7542	1.1943
	30	$N(0,1)$	0.8083	0.7526	0.4486
	20	$N(0,1)$	1.2895	0.7235	0.6550
	10	$N(0,1)$	0.5823	4.3613	0.7283
	5	$N(0,1)$	1.3314	6.4306	1.8085
II	40	$N(0,1)$	0.9641	0.6758	0.7596
	30	$N(0,1)$	1.0334	0.7864	0.7318
	20	$N(0,1)$	2.5886	0.8808	3.1920
	10	$N(0,1)$	0.9919	3.7826	1.1394
	5	$N(0,1)$	1.6796	7.6127	2.1461
III	40	$N(0,0.1)$	0.6017	0.2202	0.4931
	30	$N(0,0.1)$	1.8677	0.3725	2.0004
	20	$N(0,0.1)$	1.9294	0.3560	1.9839
	10	$N(0,0.1)$	0.4918	3.9742	0.4646
	5	$N(0,0.1)$	1.2998	8.9038	2.4679

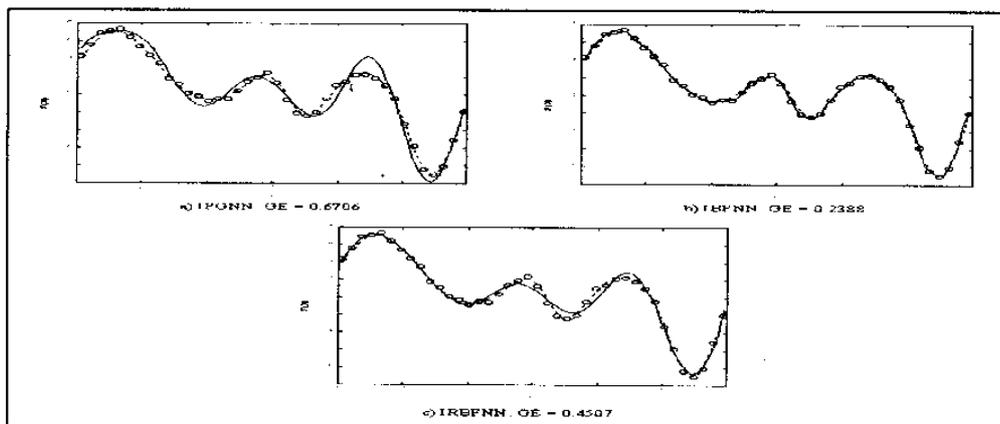
Pada Tabel 3 kita dapat melihat perbandingan kapabilitas generalisasi berdasarkan nilai variansi *noise* data pembelajaran. Nilai variansi yang diambil adalah nilai variansi yang cukup besar, yaitu lebih besar dari satu. Hasil yang diperoleh menunjukkan bahwa kapabilitas generalisasi dari PGNN selalu lebih baik dari BP maupun RBF. Dari Tabel 3 dan Tabel 2 kita dapat menarik kesimpulan bahwa PGNN memiliki kapabilitas generalisasi yang lebih baik



Gambar 3. Simulasi grafis untuk kasus III dengan lima data pembelajaran dari Tabel 3.2. Garis solid dan putus-putus adalah fungsi target dan fungsi hasil pembelajaran. o adalah data pembelajaran



Gambar 4. Simulasi grafis untuk kasus V dari Tabel 3.3. Garis solid dan putus-putus adalah fungsi target dan fungsi hasil pembelajaran. o adalah data pembelajaran.



Gambar 5. Simulasi grafis untuk kasus IV dari Tabel 3.4. Garis solid dan putus-putus adalah fungsi target dan fungsi hasil pembelajaran. o adalah data pembelajaran

**Studi Komparasi Terhadap Kapabilitas Generalisasi dari Jaringan Saraf Tiruan
Berbasis Incremental Projection Learning**
[Hendri Murfi et.al.]

ketika variansi *noise* yang cukup besar berapapun jumlah data pembelajaran. Simulasi grafis untuk kasus V diberikan pada Lampiran pada gambar 4

Tabel 3. Pernadingan kapabilitas generalisasi untuk variansi noise cukup besar

Kasus	Jumlah Data	Dist. Noise	IPGNN	IBPNN	IRBFNN
			GE	GE	GE
I	40	N(0.2)	1.3102	3.2028	4.6839
II	40	N(0.2)	1.6034	2.5719	11.5226
III	40	N(0.2)	0.8587	2.5123	1.5664
IV	40	N(0.3)	4.1020	7.1684	18.9112
V	40	N(0.3)	1.6403	4.5324	2.6182
VI	40	N(0.3)	2.5363	4.7056	8.5409
VII	40	N(0.4)	2.2749	8.1463	4.5663
VIII	40	N(0.4)	5.6786	7.9739	16.1306
IX	40	N(0.4)	5.0104	8.6923	8.5244

Kasus terakhir yang akan diperbandingkan adalah kasus dimana data pembelajaran cukup besar dan nilai variansi *noise* cukup kecil. Tabel 4 menunjukkan nilai *generalization error* untuk data pembelajaran dengan kondisi tersebut. Dari Tabel 4 kita dapat melihat bahwa kapabilitas generalisasi dari PGNN kurang baik jika dibandingkan dengan BP. Sementara terhadap RBF, PGNN dapat lebih baik atau lebih jelek tergantung karakteristik data pembelajarannya. Simulasi grafis dari kasus IV diberikan pada Gambar 5.

Tabel 4. Perbandingan kapabilitas generalisasi untuk data pembelajaran yang cukup besar dan variansi noise yang cukup kecil

Kasus	Jumlah Data	Dist. Noise	IPGNN	IBPNN	IRBFNN
			GE	GE	GE
I	40	N(0.0.1)	0.5265	0.3622	0.3673
II	40	N(0.0.1)	0.5718	0.2785	0.4741
III	40	N(0.0.1)	0.6082	0.3645	0.4056
IV	40	N(0.0.2)	0.6706	0.2388	0.4507
V	40	N(0.0.2)	0.6250	0.3099	0.4215
VI	40	N(0.0.2)	0.5761	0.2416	0.3307
VII	40	N(0.0.3)	0.7321	0.2314	1.0656
VIII	40	N(0.0.3)	0.5102	0.3278	0.5577
IX	40	N(0.0.3)	0.6315	0.2604	1.4034

Penutup

Pada makalah ini telah dilakukan studi komparasi terhadap kapabilitas generalisasi dari *projection generalizing neural networks* (PGNN) berbasis *incremental projection learning* (IPL) dalam memecahkan masalah

aproksimasi fungsi dengan jaringan saraf tiruan yang sudah umum digunakan, yaitu jaringan *on-line Back Propagation* (BP) dan *Radial Basis Function* (RBF). Berdasarkan hasil uji coba tersebut, telah didapatkan kondisi berikut ini:

- PGNN tidak selalu memberikan kapabilitas generalisasi yang lebih baik. PGNN memberikan kapabilitas generalisasi yang lebih baik ketika jumlah data pembelajaran cukup kecil atau variansi *noise* dari data pembelajaran cukup besar
- Selain dua kondisi diatas, PGNN tidak selalu memberikan kapabilitas generalisasi yang lebih baik. Bahkan untuk kondisi dimana jumlah data pembelajaran cukup besar dan variansi *noise* dari data pembelajaran cukup kecil, PGNN memberikan kapabilitas generalisasi yang lebih jelek dari BP.

Referensi

- [1] Chen, S., C.F.N. Cowan and P. M. Grant. *Orthogonal Least Squares Learning Algorithm for Radial Basis Functions Networks*. IEEE Transaction on Neural Networks, Vol. 2, No. 2, pp. 302-309, 1991
- [2] Demuth, H. and M. Beale. Neural Network Toolbox 4.0 User's Guide for Matlab. The MathWorks Inc., 2000
- [3] D.E. Rumelhart, G.E. Hinton and R.J. Williams. *Learning Internal Representation by Error Propagation*. Paralel Distributed Processing: Explorations in the Micro-structure of Cognition, pp. 318-362, The MIT Press, Cambridge, MA, 1986
- [4] D.L. Yu, J.B. Gomm, D. Williams. *A Recursive Orthogonal Least Squares Algorithm for training RBF Networks*. Neural Processing Letter, vol. 5, no. 3, pp. 167-176, 1997
- [5] Ogawa, H. *Neural Networks Learning, Generalization and Over Learning*. Proceedings of the ICIIPS'92, International Conference on Intelligent Information Processing System, Beijing, China, 1992
- [6] Sugiyama, M. and H. Ogawa. *Exact Incremental Projection Learning in Neural Networks*. IEICE Technical Report, NC98-97, pp. 149-156, 1999
- [7] Sugiyama, M. and H. Ogawa. *Incremental Projection Learning for Optimal Generalization*. Neural Networks, Vol. 14, no. 1, pp. 53-66, 2001

- [8] Sugiyama, M. and H. Ogawa. *Properties of Incremental projection learning*. Neural Networks, Vol. 14, no. 1, pp. 53-66, 2001
- [9] Sugiyama, M. and H. Ogawa. *Incremental Projection Learning for Optimal Generalization in Neural Networks*. (submitted) also as Technical Report, Department of Computer Science, Tokyo Institute of Technology, Japan
- [10] Vijayakumar, S. and H. Ogawa. *RHKS Based Functional Analysis for Exact Incremental Learning*, Neurocomputing (special issue on theoretical analysis of real valued function classes). Amsterdam: Elsevier Science, 29 (1-3), 85-113, 1998