

Penyelesaian Masalah Syarat Batas dalam Persamaan Diferensial Biasa Orde Dua dengan Menggunakan Algoritma *Shooting Neural Networks*

Dewi Erla Mahmudah¹, Ratna Dwi Christyanti², Moh. Khoridatul Huda³,
Fidia Deny Tisna Amijaya⁴

¹STMIK Widya Utama, ²Universitas Kaltara, ³Universitas Islam Raden Rahmat Malang,

⁴Universitas Mulawarman

¹dewierla@swu.ac.id, ²r_christyanti@yahoo.com, ³hudaemka@gmail.com, ⁴fidiadta@unmul.ac.id

Abstract—The problem of initial values and boundary values in differential equations is still a problem that is not simple in finding the solution. In this research we will introduce a new method to solve the problem of boundary conditions in ordinary differential equations using – method and Neural Networks Algorithm (*Shooting Neural Networks*). The results of this study indicate that the solution obtained by using the shooting method is better than the method of shooting neural networks when the guess point value on the shooting method approaches the exact value. Methods of shooting neural neural networks are best used for problems with unknown exact solutions.

Keywords—Ordinary Differential Equation, Boundary Value Problems, Shooting Method, Shooting Neural Networks

1. PENDAHULUAN

Suatu permasalahan dalam teknik dan ilmu pengetahuan alam dapat dirumuskan dalam persamaan diferensial (*differential equation*). Persamaan diferensial yang digunakan dapat berupa persamaan diferensial biasa atau persamaan diferensial parsial, orde satu atau orde dua, linear atau non linear. Persamaan tersebut dapat diselesaikan secara analitik maupun numerik. Jika penyelesaian secara analitik, maka solusi yang diperoleh berupa nilai eksak. Sedangkan dengan cara numerik, maka solusi yang diperoleh berupa suatu nilai aproksimasi (taksiran). Tetapi sebagian besar dari persamaan yang ada sulit diselesaikan secara analitik sehingga harus didekati secara numerik.

Metode pendekatan numerik dari persamaan diferensial biasa tergantung pada keadaan awal yang diberikan. Jika keadaan awal diberikan pada titik tunggal tertentu, maka metode numerik yang digunakan adalah metode untuk penyelesaian masalah nilai awal. Tidak semua aplikasi persamaan diferensial mempunyai syarat awal, tetapi memberikan syarat batas pada lebih dari satu titik sehingga metode yang digunakan adalah penyelesaian masalah syarat batas.

Ada beberapa metode yang digunakan untuk menyelesaikan masalah syarat batas (*boundary value problems/BVPs*) pada persamaan diferensial biasa. Salah satu metode yang sering digunakan yaitu *shooting method*. Metode *shooting method* dapat

diterapkan pada masalah linier maupun non linier. Metode *shooting method* merubah suatu persamaan diferensial biasa orde dua dengan syarat batas menjadi suatu sistem persamaan diferensial biasa orde satu dengan nilai awal yang dapat diselesaikan dengan metode penyelesaian masalah nilai awal. Dengan demikian, didapatkan dua persamaan diferensial biasa orde satu. Selanjutnya, menentukan nilai awal untuk dua persamaan diferensial orde satu tersebut dengan cara menebak ('menembak'/*shooting*) suatu nilai. Dengan nilai tebakan tersebut diharapkan nilai batas baru mendekati nilai batas sebenarnya. Untuk mendapatkan suatu nilai batas baru dapat digunakan metode Runge-Kutta orde empat.

Untuk mendapatkan hasil yang akurat dari metode *shooting*, ditentukan minimal dua tebakan nilai yang nilainya mendekati solusi eksaknya. Akan tetapi hal ini sulit dilakukan karena metode *shooting* digunakan ketika tidak mengetahui solusi eksaknya. Jadi cara lain untuk mendekati solusi eksaknya adalah dengan menebak dua nilai sembarang kemudian dari hasil tebakan itu dilakukan metode *shooting* lagi dengan memanfaatkan nilai hasil tebakan untuk menghasilkan hasil tebakan baru begitu seterusnya. Akan tetapi cara ini membutuhkan waktu komputasi yang lebih lama, jika nilai tebakan awal sangat jauh dari solusi eksak dan hasil akhir dari metode *shooting* kadangkala masih menghasilkan *error*. Jadi, disini akan diperkenalkan metode baru untuk menyelesaikan masalah syarat batas dalam persamaan diferensial biasa yaitu menggunakan metode *shooting* dan algoritma jaringan syaraf (*Neural Networks Algorithm*) yang diberi nama algoritma *shooting neural networks*. Dengan harapan, dapat menghasilkan *error* yang semakin kecil atau dengan kata lain mendapatkan hasil yang semakin akurat. Pada penelitian ini akan dibahas bagaimana menyelesaikan masalah syarat batas dalam persamaan diferensial biasa orde dua dengan menggunakan algoritma *Shooting Neural Networks*.

2. METODE PENELITIAN

Metode penelitian yang digunakan dalam penelitian ini adalah studi literatur yaitu dengan menggunakan metode *shooting neural networks* yang merupakan metode penebakan turunan pertama dari suatu masalah

nilai batas dengan menggunakan penggabungan metode *shooting* dan algoritma *neural networks*. Metode ini ditemukan oleh Kais Ismail Ibraheem dan Bashir M. Khalaf pada tahun 2010. Secara umum, metode ini dibagi menjadi 3 tahap :

1. *Shooting Technique* untuk menentukan nilai inputan pada *neural networks*.
2. *Neural Network Technique* untuk melakukan proses pembelajaran dan mendapatkan pola dari pembelajaran tersebut.
3. Input nilai batas akhir yang diketahui untuk menentukan turunan pertama pada batas awal.

Secara umum, algoritma *Shooting Neural Networks* dapat dituliskan sebagai berikut; Misalkan diketahui BVP dengan dua titik :

$$\frac{d^2 y}{dx^2} = f(x, y, \frac{dy}{dx}), y(a) = \alpha, y(b) = \beta, x \in [a, b]$$

Masalah diatas dapat disederhanakan menjadi sistem sebagai berikut :

$$\begin{aligned} \frac{dy_1}{dx} &= y_2 \\ \frac{dy_2}{dx} &= f\left(x, y, \frac{dy}{dx}\right), y_1(a) = \alpha, y_1(b) = \beta \end{aligned}$$

Untuk mengintegrasikan sistem diatas pada interval $[a, b]$ dibutuhkan sebuah nilai $y_2(a)$ yang masih belum diketahui.

Untuk mendapatkan nilai ini, dilakukan proses seperti dibawah ini :

1. Shooting Technique

Proses 1 Menebak suatu nilai $y_2(a) = s_1$ dan mengintegrasikan system pada interval $[a, b]$ sehingga didapatkan $y_1(b) = m_1$

Proses 2 Menebak suatu nilai $y_2(a) = s_2$ dan mengintegrasikan system pada interval $[a, b]$ sehingga didapatkan $y_1(b) = m_2$ dan seterusnya.

⋮

Proses n Menebak suatu nilai $y_2(a) = s_n$ dan mengintegrasikan system pada interval $[a, b]$ sehingga didapatkan $y_1(b) = m_n$

sehingga didapatkan Tabel 1 di bawah ini :

Tabel 1 Tebakan Nilai pada *Shooting Technique*

$s_i = y_2(a)$	s_1	s_2	s_3	...	s_n
$m_i = y_1(b)$	m_1	m_2	m_3	...	m_n

Tabel 1 dapat dituliskan sebagai Tabel 2 berikut :

Tabel 2 Nilai input dan Target

Input	m_1	m_2	m_3	...	m_n
Target	s_1	s_2	s_3	...	s_n

kemudian dengan menggunakan *neural networks* akan dicari pola data dari input ke target. Sehingga apabila diketahui nilai $y_1(b) = \beta$ (input) maka dapat ditentukan nilai $y_2(a)$ (target).

2. Neural Networks Technique

Langkah 0 Inisialisasi bobot (v dan w)

Langkah 1 Ketika kondisi berhenti belum terpenuhi, lakukan langkah 2-9

Langkah 2 Untuk setiap pasangan data, lakukan langkah 3-8

Feed Forward

Langkah 3 Masing-masing neuron masukan ($x_i, i = 1, \dots, n$). Masukkan sinyal x_i dan menghantarkan sinyal ke layer selanjutnya (*hidden units*).

Langkah 4 Masing-masing neuron pada *hidden units* ($z_j, j = 1, \dots, r$), menjumlahkan sinyal masukan dengan persamaan :

$$z_{in_j} = v_{0j} + \sum_{i=1}^n m_j v_{ij}$$

dan menghitung fungsi aktivasi untuk menghitung neuron *hidden units*,

$z_j = f(z_{in_j})$ yang mana fungsi aktivasinya menggunakan fungsi sigmoid.

Langkah 5 Masing-masing neuron keluaran ($y_k, k = 1, \dots, r$), menjumlahkan sinyal masukan sinyal neuron pada *hidden units* dengan persamaan :

$$y_{in_k} = w_{0k} + \sum_{j=1}^r z_j w_{jk}$$

dan menghitung fungsi aktivasinya untuk menghitung neuron keluaran,

$y_k = f(y_{in_k})$ yang mana fungsi aktivasinya menggunakan fungsi linier.

Backpropagation

Langkah 6 Masing-masing neuron keluaran ($y_k, k = 1, \dots, r$), menerima pola target terhadap pola neuron pada *hidden units*, kemudian menghitung informasi kesalahannya,

$$\delta_k = (t_i - y_k) f'(y_{in_k})$$

menghitung koreksi bobot w

$$\Delta w_{jk} = \alpha \delta_k z_j$$

menghitung koreksi bias,

$$\Delta w_{0k} = \alpha \delta_k$$

- Langkah 7 dan mengirim δ_k ke unit layer sebelumnya.
Masing-masing neuron pada hidden units ($z_j, j = 1, \dots, r$), menjumlahkan masukan deltanya (dari unit selanjutnya), $\delta_{in_j} = \sum_{k=1}^r \delta_k w_{jk}$ kalikan dengan turunan fungsi aktivasi untuk menghitung informasi kesalahan, $\delta_j = \delta_{in_j} f'(z_{in_j})$ menghitung koreksi bobot, $\Delta w_{ij} = \alpha \delta_j x_i$ menghitung koreksi bias, $\Delta v_{0j} = \alpha \delta_j$
- Langkah 8 Masing-masing neuron keluaran ($y_k, k = 1, \dots, r$) update bias dan bobotnya ($j = 0, \dots, r$),
 $w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk}$
Masing-masing neuron pada hidden units ($z_j, j = 1, \dots, r$) update bias dan bobotnya ($i = 0, \dots, r$),
 $v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij}$
- Langkah 9 Tes kondisi berhenti.
(Ibraheem dan Khalaf, 2011)
3. Input nilai $y_1(b) = \beta$ untuk mendapatkan nilai $y_2(a)$.

3. HASIL DAN PEMBAHASAN

3.1 Permasalahan I

Masalah yang akan diselesaikan adalah masalah syarat batas persamaan diferensial biasa orde dua, yaitu

$$\frac{d^2 y}{dx^2} = -\frac{dy}{dx} + 2y \quad (3.1)$$

$$y(0) = 1, \quad y(1) = e$$

$$f_1(x_n, u_n, v_n) = v(x_n) \Rightarrow f_1(x_0, u_0, v_0) = v(x_0 = 0) = 1$$

$$f_2(x_n, u_n, v_n) = -v(x_n) + 2u(x_n) \Rightarrow f_2(x_0, u_0, v_0) = -v(x_0 = 0) + 2u(x_0 = 0) = -1 + 2.1 = 1$$

$$k_1 = hf_1(x_0, u_0, v_0) = 0,1. v_0 = 0,1.1 = 0,1$$

$$l_1 = hf_2(x_0, u_0, v_0) = 0,1.(-v_0 + 2u_0) = 0,1.1 = 0,1$$

$$k_2 = hf_1\left(x_0 + \frac{h}{2}, u_0 + \frac{k_1}{2}, v_0 + \frac{l_1}{2}\right) = 0,1.\left(v_0 + \frac{l_1}{2}\right) = 0,1.1,05 = 0,105$$

diperoleh solusi eksak $y = e^x$.

Bukti:

$$\frac{d^2 y}{dx^2} + \frac{dy}{dx} - 2y = 0$$

$$\Leftrightarrow (D^2 + D - 2)y = 0 \Rightarrow f(D) = D^2 + D - 2$$

$$f(m) = m^2 + m - 2$$

Persamaan karakteristik:

$$f(m) = 0 \Leftrightarrow m^2 + m - 2 = 0$$

$$\Leftrightarrow (m + 2)(m - 1) = 0$$

$$\Leftrightarrow m = -2 \vee m = 1$$

Solusi umum:

$$y = C_1 e^x + C_2 e^{-2x} \quad (3.2)$$

Jika syarat $y(0) = 1$ dan $y(1) = e$ disubstitusikan ke persamaan (3.2), maka diperoleh $C_1 = 1$ dan $C_2 = 0$. Jadi solusi eksaknya adalah $y = e^x$. ■

Dari solusi eksak $y = e^x$ maka didapat

$$y' = e^x \quad (3.3)$$

dan

$$y'' = e^x \quad (3.4)$$

Substitusikan persamaan (3.3) dan persamaan (3.4) ke persamaan (3.1), diperoleh

$$e^x = -e^x + 2e^x$$

$$e^x + e^x - 2e^x = 0$$

Kemudian uji syarat batas untuk $x = 0$, diperoleh $y(0) = e^0 = 1$. Selanjutnya uji syarat batas untuk $x = 1$, diperoleh $y(1) = e^1 = e$. Terbukti bahwa persamaan (3.1) mempunyai solusi yaitu $y = e^x$.

Persamaan diferensial pada persamaan (3.1) dapat diselesaikan secara numerik dengan metode *shooting*. Untuk menyelesaikannya, persamaan (3.1) terlebih dahulu diubah ke bentuk sistem persamaan diferensial orde satu, dengan memisalkan $y(x) = u(x)$ sehingga diperoleh dua sistem persamaan diferensial orde satu, yaitu

$$\frac{dy(x)}{dx} = y'(x) = u'(x) = v(x)$$

$$\frac{dv(x)}{dx} = y''(x) = -y'(x) + 2y(x)$$

$$= -v(x) + 2u(x)$$

Diketahui nilai awal $y(0) = u(0) = 1$. Dengan menebak suatu nilai $v(0) = y'(0) = s = 1$, dan dengan menggunakan metode Runge Kutta orde empat dengan $h = 0,1$, diperoleh

$$l_2 = hf_2\left(x_0 + \frac{h}{2}, u_0 + \frac{k_1}{2}, v_0 + \frac{l_1}{2}\right) = 0,1 \cdot \left(-\left(v_0 + \frac{l_1}{2}\right) + 2\left(u_0 + \frac{k_1}{2}\right)\right) = 0,1 \cdot (-1,05 + 2,1,05) \\ = 0,105$$

$$k_3 = hf_1\left(x_0 + \frac{h}{2}, u_0 + \frac{k_2}{2}, v_0 + \frac{l_2}{2}\right) = 0,1 \cdot \left(v_0 + \frac{l_2}{2}\right) = 0,1 \cdot 1,0525 = 0,10525$$

$$l_3 = hf_2\left(x_0 + \frac{h}{2}, u_0 + \frac{k_2}{2}, v_0 + \frac{l_2}{2}\right) = 0,1 \cdot \left(-\left(v_0 + \frac{l_2}{2}\right) + 2\left(u_0 + \frac{k_2}{2}\right)\right) \\ = 0,1 \cdot (-1,0525 + 2,1,0525) = 0,1 \cdot 1,105 = 0,10525$$

$$k_4 = hf_1(x_0 + h, u_0 + k_3, v_0 + l_3) = 0,1 \cdot (v_0 + l_3) = 0,1 \cdot 1,10525 = 0,110525$$

$$l_4 = hf_2(x_0 + h, u_0 + k_3, v_0 + l_3) = 0,1 \cdot (-v_0 + l_3) + 2(u_0 + k_3) \\ = 0,1 \cdot (-1,10525 + 2(1,10525)) = 0,110525$$

Substitusikan hasil k_1, k_2, k_3, k_4 sehingga diperoleh

$$u(0,1) = u_1 = u_0 + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\ = 1 + \frac{1}{6}(0,1 + 2,0,105 + 2,0,10525 + 0,110525) \\ = 1 + \frac{1}{6}(0,1 + 0,21 + 0,2105 + 0,110525) \\ = 1 + \frac{1}{6}(0,631025) \\ = 1 + 0,10517083 \\ = 1,10517083$$

Selanjutnya substitusikan hasil l_1, l_2, l_3, l_4 sehingga diperoleh

$$v(0,1) = v_1 = v_0 + \frac{1}{6}(l_1 + 2l_2 + 2l_3 + l_4) \\ = 1 + \frac{1}{6}(0,1 + 2,0,105 + 2,0,10525 + 0,110525) \\ = 1 + \frac{1}{6}(0,1 + 0,21 + 0,2105 + 0,110525) \\ = 1 + \frac{1}{6}(0,631025) \\ = 1 + 0,10517083 \\ = 1,10517083$$

Kemudian dilakukan perulangan sampai iterasi ke-10 sehingga diperoleh nilai $u(1)$ seperti pada Tabel 3 berikut ini:

Tabel 3 Nilai $u(1)$ pada Permasalahan I

x_n	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
$u(x_n)$	1,2214	1,3499	1,4918	1,6487	1,8221	2,0138	2,2255	2,4596	2,7183

Dari Tabel 3 di atas dapat diketahui bahwa pada iterasi ke-10 yaitu $x_{10} = 1$ diperoleh nilai $(1) = 2,7183$. Hal ini menunjukkan bahwa perhitungan secara numerik dari persamaan (3.1) dengan menggunakan metode *shooting* didapatkan nilai $u(1) = y(1)$ yang mendekati nilai batas sebenarnya yaitu $y(1) = e = 2,7183$.

Dari metode *shooting* didapatkan suatu nilai $y(1) = 2,7183$ yang nantinya akan digunakan untuk mencari suatu nilai $y'(0) = s$ dan s tersebut akan dijadikan nilai tebakan dalam prosedur *neural networks algorithm*. Dalam hal ini, nilai tebakan adalah $s = 1$ dan ternyata menghasilkan solusi yang akurat. Hal ini dimungkinkan karena tebakan yang terjadi secara acak tepat mengenai solusi eksaknya. Akan tetapi dalam kondisi nyata tebakan yang tepat ini jarang didapat apalagi jika solusi eksaknya bernilai desimal. Selanjutnya akan dicari nilai $y'(0)$ menggunakan metode *shooting neural networks*. Sesuai dengan algoritma *shooting neural networks*, langkah

awal yang harus dilakukan adalah melakukan metode *shooting* dengan beberapa nilai tebakan ($y'(0)$) kemudian melakukan metode runge kutta orde 4 untuk mendapatkan nilai pada batas akhirnya ($y(1)$). Misalkan diketahui lima nilai tebakan dan nilai dibatas akhirnya seperti pada Tabel 4 berikut:

Tabel 4 Nilai Tebakan dan Nilai Batas pada Permasalahan I

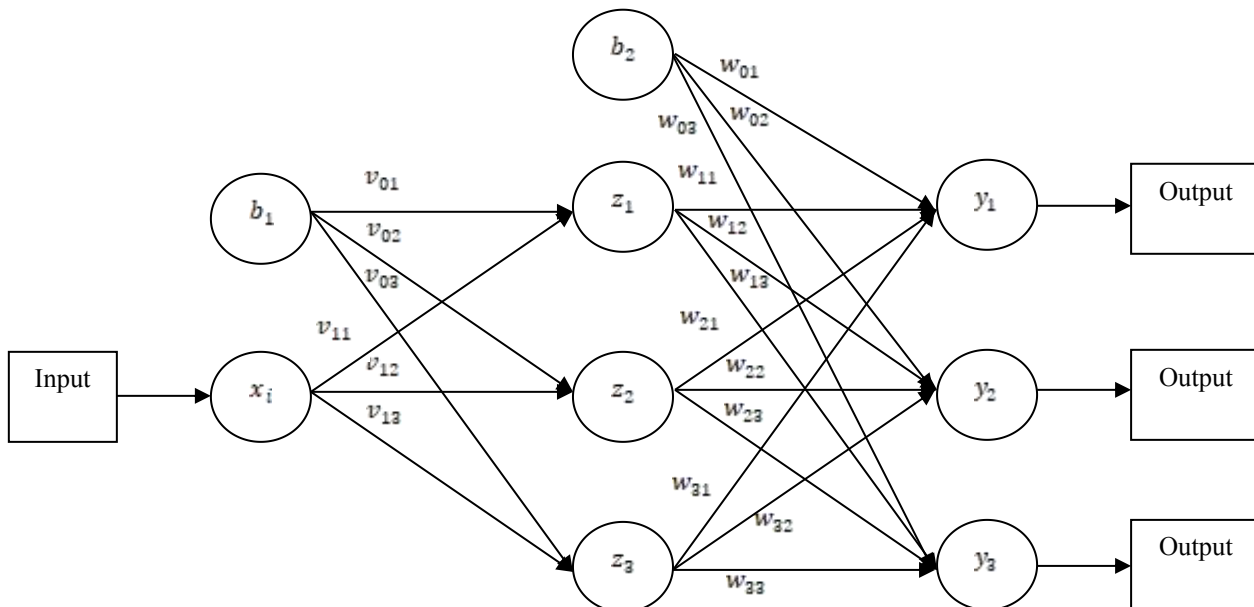
$y'(0)$	-1	0	1	2	3
$y(1)$	0.9963	1.8573	2.7183	3.5793	4.4404

Selanjutnya Tabel 4 di atas digunakan sebagai tabel pembelajaran *neural networks*. Tabel ini nanti digunakan untuk mengetahui pola dari *neural networks*. Karena dari soal yang diketahui adalah $y(1) = e$ dan yang ingin diketahui adalah nilai $y'(0)$ maka dalam menginputkan ke *neural networks* tabel 4 dapat dituliskan menjadi Tabel 5 berikut:

Tabel 5 Nilai Input dan Target pada Permasalahan I

Input ($y(1)$)	0.9963	1.8573	2.7183	3.5793	4.4404
Target ($y'(0)$)	-1	0	1	2	3

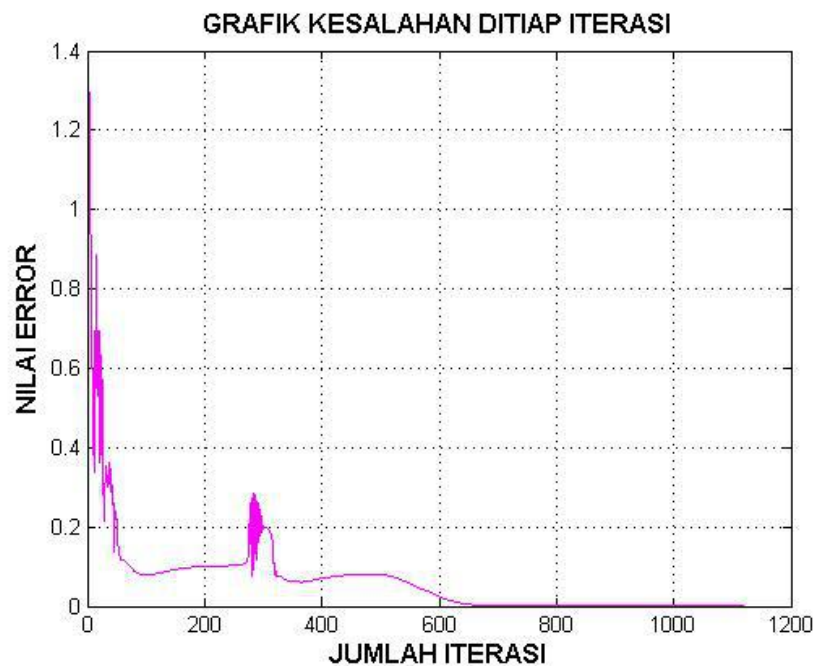
Sesuai dengan algoritma *shooting neural networks*, arsitektur *neural networks* yang akan digunakan dapat dilihat pada Gambar 1 berikut:

Gambar 1 Arsitektur Neural Networks dengan 1 Hidden Layer, 1 Input x , 3 Node z , dan 3 Node y

Tabel 6 Parameter Neural Networks

Jumlah node layer 1	Jumlah node output	Learning rate (α)	Kesalahan maksimum
5		1.5	0.0001

Dengan menggunakan parameter *neural networks* pada Tabel 6, maka didapatkan grafik kesalahan pembelajaran seperti pada Gambar 2 berikut:



Gambar 2 Grafik Kesalahan Pembelajaran Permasalahan I di Setiap Iterasinya

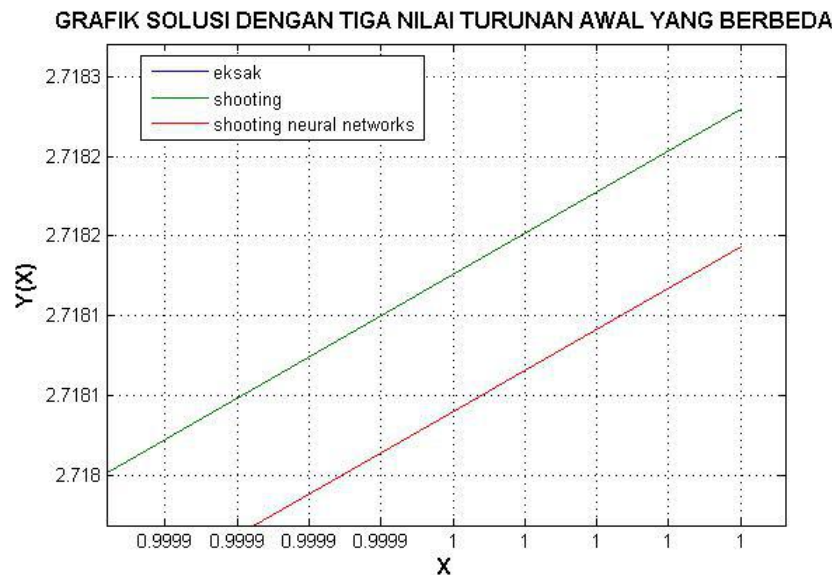
Dari grafik dapat dilihat nilai error pada iterasi terakhir adalah kurang dari 0,0001. Artinya diperoleh nilai bobot v dan w yang sudah optimal sehingga ketika diinputkan nilai 0.9963 maka nilai keluaran yang didapat adalah -1,0001. Nilai bobot v dan w tersebut digunakan untuk menentukan nilai $y'(0)$. Dengan menginputkan nilai batas akhir yang sudah diketahui di permasalahan I, yaitu $y(1) = e$ maka didapatkan $y'(0) = 0,9999$. Dari ketiga perhitungan di atas, didapatkan hasil seperti pada Tabel 7 berikut:

Tabel 7 Syarat Nilai pada Permasalahan I

$y'(0)$ eksak	$y'(0)$ metode shooting	$y'(0)$ metode shooting neural networks
1	1	0,9999

Dengan menggunakan tiga nilai syarat di atas dan metode runge kutta orde 4 diperoleh perbandingan tiga grafik yang dapat dilihat pada Gambar 3 di bawah ini:

Gambar 3 Grafik Perbandingan Tiga Nilai Fungsi dengan Syarat $y'(0)$ Eksak, $y'(0)$ Metode Shooting, $y'(0)$ Metode Shooting Neural Networks



Gambar 4 Perbesaran Gambar 3

Dari Gambar 4 dapat dilihat bahwa grafik *shooting* sama dengan grafik eksaknya. Sedangkan grafik *shooting neural networks* berada di bawah grafik *shooting* dan grafik eksak. Hal ini disebabkan karena nilai $y(0)$ yang digunakan pada grafik *shooting* sama dengan nilai eksaknya sedangkan nilai $y'(0)$ yang digunakan pada *shooting neural networks* nilainya hanya mendekati nilai $y(0)$ eksak. Sehingga $y(1)$ yang didapatkan dari *shooting neural networks* tidak sama dengan $y(1)$ eksak, tetapi hanya mendekati. Dari Permasalahan I dapat disimpulkan bahwa metode *shooting* memiliki nilai yang lebih akurat daripada nilai yang didapat dari *shooting neural networks*. Hal ini disebabkan karena tebakan awal nilai *shooting* tepat mengenai solusi eksaknya.

3.2 Permasalahan II

Masalah yang akan diselesaikan adalah masalah syarat batas persamaan diferensial biasa orde dua, yaitu:

$$\frac{d^2y}{dx^2} = \frac{2y}{x^2} - \frac{1}{x} \quad (3.5)$$

$$y(2) = y(3) = 0$$

Diperoleh solusi eksak:

$$y = \frac{1}{38} \left(19x - 5x^2 - \frac{36}{x} \right)$$

Bukti:

Setiap ruas dari persamaan (3.5) dikalikan dengan x^2 , diperoleh

$$x^2 \frac{d^2y}{dx^2} - 2y = -x \quad (3.6)$$

Dengan substitusi

$$x = e^t \Leftrightarrow t = \ln x \quad (3.7)$$

$$\frac{dy}{dx} = \frac{dy}{dt} \frac{dt}{dx} = \frac{dy}{dt} \frac{d(\ln x)}{dx} = \frac{1}{x} \frac{dy}{dt} \quad (3.8)$$

dan

$$\begin{aligned} \frac{d^2y}{dx^2} &= \frac{1}{x} \frac{d}{dx} \left(\frac{dy}{dt} \right) + \frac{dy}{dt} \frac{d}{dx} \left(\frac{1}{x} \right) \\ &= \frac{1}{x} \frac{d}{dt} \left(\frac{dy}{dt} \right) \frac{dt}{dx} - \frac{1}{x^2} \frac{dy}{dt} \\ &= \frac{1}{x} \frac{d^2y}{dt^2} \frac{dt}{dx} - \frac{1}{x^2} \frac{dy}{dt} \\ &= \frac{1}{x} \frac{d^2y}{dt^2} \frac{1}{x} - \frac{1}{x^2} \frac{dy}{dt} \\ &= \frac{1}{x^2} \left(\frac{d^2y}{dt^2} - \frac{dy}{dt} \right) \end{aligned} \quad (3.9)$$

ke persamaan (3.6), diperoleh

$$\begin{aligned} x^2 \left(\frac{1}{x^2} \left(\frac{d^2y}{dt^2} - \frac{dy}{dt} \right) \right) - 2y &= -e^t \\ \frac{d^2y}{dt^2} - \frac{dy}{dt} - 2y &= -e^t \quad (3.10) \\ \Leftrightarrow (D^2 - D - 2)y &= 0 \Rightarrow f(D) = D^2 - D - 2 \\ f(m) &= m^2 - m - 2 \end{aligned}$$

Persamaan karakteristik:

$$\begin{aligned} f(m) = 0 &\Leftrightarrow m^2 - m - 2 = 0 \\ \Leftrightarrow (m - 2)(m + 1) &= 0 \\ \Leftrightarrow m = 2 \vee m = -1 \end{aligned}$$

Solusi homogen:

$$y_c = C_1 e^{2t} + C_2 e^{-t}$$

Misalkan solusi khusus:

$$\begin{aligned} y_p &= A e^t \\ y_p' &= A e^t \\ y_p'' &= A e^t \end{aligned} \quad (3.11)$$

Substitusikan persamaan (3.11) ke persamaan (3.10), diperoleh

$$\begin{aligned} A e^t - A e^t - 2A e^t &= -e^t \\ -2A e^t &= -e^t \\ A &= \frac{1}{2} \end{aligned}$$

Sehingga solusi khususnya $y_p = \frac{1}{2} e^t$.

Jadi solusi umumnya:

$$y = y_c + y_p$$

$$= C_1 e^{2t} + C_2 e^{-t} + \frac{1}{2} e^t \quad (3.12)$$

Substitusi persamaan (3.7) ke (3.12) sehingga diperoleh solusi umum

$$y(x) = C_1 x^2 + \frac{C_2}{x} + \frac{1}{2} x \quad (3.13)$$

Jika syarat $y(2) = y(3) = 0$ disubstitusikan ke persamaan (3.13), maka diperoleh $C_1 = -\frac{5}{38}$ dan $C_2 = -\frac{36}{38}$.

Substitusikan C_1 dan C_2 ke persamaan (3.13), sehingga diperoleh solusi eksaknya yaitu

$$y(x) = -\frac{5}{38} x^2 - \frac{36}{38x} + \frac{1}{2} x$$

Dengan menebak suatu nilai $s = 0,1$, dan dengan menggunakan metode Runge Kutta orde empat dengan $h = 0,1$, diperoleh

$$f_1(x_n, u_n, v_n) = v(x_n) \Rightarrow f_1(x_0, u_0, v_0) = v(x_0 = 2) = 0,1$$

$$f_2(x_n, u_n, v_n) = \frac{2}{x^2} u(x_n) - \frac{1}{x} \Rightarrow f_1(x_0, u_0, v_0) = \frac{2}{x^2} u_0 - \frac{1}{x} = \frac{2}{4} \cdot 0 - \frac{1}{2} = -0,5$$

$$k_1 = hf_1(x_0, u_0, v_0) = 0,1 \cdot 0,1 = 0,01$$

$$l_1 = hf_2(x_0, u_0, v_0) = 0,1 \cdot (-0,5) = -0,05$$

$$k_2 = hf_1\left(x_0 + \frac{h}{2}, u_0 + \frac{k_1}{2}, v_0 + \frac{l_1}{2}\right) = 0,1 \cdot 0,075 = 0,0075$$

$$l_2 = hf_2\left(x_0 + \frac{h}{2}, u_0 + \frac{k_1}{2}, v_0 + \frac{l_1}{2}\right) = 0,1 \cdot \left(-\left(v_0 + \frac{l_1}{2}\right) + 2\left(u_0 + \frac{k_1}{2}\right)\right) = 0,1 \cdot (-0,48542534)$$

$$= -0,048542534$$

$$k_3 = hf_1\left(x_0 + \frac{h}{2}, u_0 + \frac{k_2}{2}, v_0 + \frac{l_2}{2}\right) = 0,1 \cdot 0,075728733 = 0,0075728733$$

$$l_3 = hf_2\left(x_0 + \frac{h}{2}, u_0 + \frac{k_2}{2}, v_0 + \frac{l_2}{2}\right) = 0,1 \cdot \left(\frac{2}{(2,05)^2} \cdot 0,00375 - \frac{1}{2,05}\right) = 0,1 \cdot (-0,486020276)$$

$$= -0,0486020276$$

$$k_4 = hf_1(x_0 + h, u_0 + k_3, v_0 + l_3) = 0,1 \cdot (v_0 + l_3) = 0,1 \cdot (0,1 - 0,0486020276)$$

$$= 0,00513979724$$

$$l_4 = hf_2(x_0 + h, u_0 + k_3, v_0 + l_3) = 0,1 \cdot \left(\frac{2}{(2,1)^2} \cdot 0,0075728733 - \frac{1}{2,1}\right) = 0,1 \cdot (-0,472756066)$$

$$= -0,0472756066$$

Substitusikan hasil k_1, k_2, k_3, k_4 sehingga diperoleh

$$u(2,1) = u_1 = u_0 + \frac{1}{6} (k_1 + 2k_2 + 2k_3 + k_4)$$

$$= 0 + \frac{1}{6} (0,03 + 0,055 + 0,055193337 + 0,0251873884)$$

$$= \frac{1}{6} (0,165380725)$$

$$= 0,027563454$$

Substitusikan hasil l_1, l_2, l_3, l_4 sehingga diperoleh

$$v(2,1) = v_1 = v_0 + \frac{1}{6} (l_1 + 2l_2 + 2l_3 + l_4)$$

$$= 0,3 + \frac{1}{6} (-0,05 - 0,096133254 - 0,09625223 - 0,046367498)$$

$$= -\frac{5}{38} x^2 - \frac{36}{38x} + \frac{19}{38} x$$

$$y(x) = \frac{1}{38} \left(19x - 5x^2 - \frac{36}{x} \right). \blacksquare$$

Persamaan diferensial pada persamaan (3.1) dapat diselesaikan secara numerik dengan metode *shooting*. Untuk menyelesaikannya, persamaan (3.1) terlebih dahulu diubah ke bentuk sistem persamaan diferensial orde satu, dengan memisalkan $y(x) = u(x)$ sehingga diperoleh dua sistem persamaan diferensial orde satu, yaitu

$$\frac{dy(x)}{dx} = y'(x) = u'(x) = v(x)$$

$$\frac{dv(x)}{d(x)} = y''(x) = \frac{2}{x^2} y - \frac{1}{x} = \frac{2}{x^2} u(x) - \frac{1}{x}$$

Didefinisikan nilai awal baru yaitu:

$$y(2) = 0 \quad y'(2) = s$$

$$\begin{aligned}
&= 0,3 + \frac{1}{6}(-0,288752982) \\
&= 0,3 - 0,048125497 \\
&= 0,251874503
\end{aligned}$$

Kemudian dilakukan perulangan sampai iterasi ke-10 sehingga diperoleh nilai $u(3)$ seperti pada Tabel 8 berikut ini:

Tabel 8 Nilai $u(3)$ dengan Tebakan Nilai $s = 0,1$

x_n	2,2	2,3	2,4	2,5	2,6	2,7	2,8	2,9	3
$u(x_n)$	0,0104	0,0087	0,0027	-0,0075	-0,0217	-0,0398	-0,0617	-0,0874	-0,1167

Dari Tabel 8 di atas dapat diketahui bahwa pada iterasi ke-10 yaitu $x_{10} = 1$ diperoleh nilai $u(3) = -0,1167$. Hal ini menunjukkan bahwa perhitungan secara numerik dari persamaan (3.5) dengan menggunakan metode *shooting* dengan menebak $s = 0,1$, didapatkan nilai $u(3) = y(3)$ yang belum mendekati nilai batas sebenarnya yaitu $y(3) = 0$.

Karena pada saat nilai tebakan $s = 0,1$ belum mendekati nilai batas sebenarnya, maka selanjutnya akan dilakukan prosedur perhitungan secara numerik untuk persamaan (3.5) dengan nilai tebakan $s = 0,3$.

Dengan menggunakan metode Runge Kutta orde empat, dengan $h = 0.1$ maka

$$\begin{aligned}
f_1(x_n, u_n, v_n) &= v(x_n) \Rightarrow f_1(x_0, u_0, v_0) = v(x_0 = 2) = 0,3 \\
f_2(x_n, u_n, v_n) &= \frac{2}{x^2} u(x_n) - \frac{1}{x} \Rightarrow f_1(x_0, u_0, v_0) = \frac{2}{x^2} u_0 - \frac{1}{x} = \frac{2}{4} \cdot 0 - \frac{1}{2} = -0,5 \\
k_1 &= hf_1(x_0, u_0, v_0) = 0,1 \cdot 0,3 = 0,03 \\
l_1 &= hf_2(x_0, u_0, v_0) = 0,1 \cdot (-0,5) = -0,05 \\
k_2 &= hf_1\left(x_0 + \frac{h}{2}, u_0 + \frac{k_1}{2}, v_0 + \frac{l_1}{2}\right) = 0,1 \cdot (0,3 - 0,025) = 0,1 \cdot 0,275 = 0,0275 \\
l_2 &= hf_2\left(x_0 + \frac{h}{2}, u_0 + \frac{k_1}{2}, v_0 + \frac{l_1}{2}\right) = 0,1 \cdot \left(-\left(v_0 + \frac{l_1}{2}\right) + 2\left(u_0 + \frac{k_1}{2}\right)\right) \\
&= 0,1 \cdot \left(\frac{2}{(2,05)^2} \cdot 0,015 - \frac{1}{2,05}\right) = 0,1 \cdot (-0,48066627) = -0,048066627 \\
k_3 &= hf_1\left(x_0 + \frac{h}{2}, u_0 + \frac{k_2}{2}, v_0 + \frac{l_2}{2}\right) = 0,1 \cdot (0,3 - 0,024033313) = 0,1 \cdot 0,275966687 \\
&= 0,0275966687 \\
l_3 &= hf_2\left(x_0 + \frac{h}{2}, u_0 + \frac{k_2}{2}, v_0 + \frac{l_2}{2}\right) = 0,1 \cdot \left(\frac{2}{(2,05)^2} \cdot 0,01375 - \frac{1}{2,05}\right) = 0,1 \cdot (-0,481261154) \\
&= -0,0481261154 \\
k_4 &= hf_1(x_0 + h, u_0 + k_3, v_0 + l_3) = 0,1 \cdot (v_0 + l_3) = 0,1 \cdot (0,3 - 0,0481261154) \\
&= 0,1 \cdot 0,251873884 = 0,0251873884 \\
l_4 &= hf_2(x_0 + h, u_0 + k_3, v_0 + l_3) = 0,1 \cdot \left(\frac{2}{(2,1)^2} \cdot 0,0275966687 - \frac{1}{2,1}\right) = 0,1 \cdot (-0,46367498) \\
&= -0,046367498
\end{aligned}$$

Substitusikan hasil k_1, k_2, k_3, k_4 sehingga diperoleh

$$\begin{aligned}
u(2,1) &= u_1 = u_0 + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\
&= 0 + \frac{1}{6}(0,03 + 0,055 + 0,055193337 + 0,0251873884) \\
&= \frac{1}{6}(0,165380725) \\
&= 0,027563454
\end{aligned}$$

Substitusikan hasil l_1, l_2, l_3, l_4 sehingga diperoleh

$$\begin{aligned}
v(2,1) &= v_1 = v_0 + \frac{1}{6}(l_1 + 2l_2 + 2l_3 + l_4) \\
&= 0,3 + \frac{1}{6}(-0,05 - 0,096133254 - 0,09625223 - 0,046367498)
\end{aligned}$$

$$= 0,3 + \frac{1}{6}(-0,288752982)$$

$$= 0,251874503$$

Kemudian dilakukan perulangan sampai iterasi ke-10 sehingga diperoleh nilai $u(3)$ seperti pada Tabel 9 berikut:

Tabel 9 Nilai $u(3)$ dengan Tebakan Nilai $s = 0,3$

x_n	2,2	2,3	2,4	2,5	2,6	2,7	2,8	2,9	3
$u(x_n)$	0,0505	0,0691	0,0836	0,0942	0,1011	0,1044	0,1044	0,1010	0,0944

Dari Tabel 9 di atas dapat diketahui bahwa pada iterasi ke-10 yaitu $x_{10} = 1$ diperoleh nilai $u(3) = 0,0944$. Hal ini menunjukkan bahwa perhitungan secara numerik dari persamaan (3.5) dengan menggunakan metode *shooting* dan dengan menebak $s = 0,3$, didapatkan $u(3) = y(3)$ yang belum mendekati nilai batas sebenarnya yaitu $y(3) = 0$. Karena tebakan $s = 0,1$ dan $s = 0,3$ belum mendekati nilai batas sebenarnya, maka dilakukan pendekatan dari interpolasi linear untuk menentukan suatu tebakan s yang akurat sebagai berikut:

$\alpha_0 = 0,1$ dan $\alpha_1 = 0,3$

$$s = \alpha_{k+1} = \alpha_{k-1} + (\alpha_k - \alpha_{k-1}) \left(\frac{y_b - y(\alpha_{k-1}; b)}{y(\alpha_k; b) - y(\alpha_{k-1}; b)} \right)$$

$$s = \alpha_2 = \alpha_0 + (\alpha_1 - \alpha_0) \left(\frac{y(1) - y(\alpha_0; 1)}{y(\alpha_1; 1) - y(\alpha_0; 1)} \right)$$

$$= 0,1 + (0,2) \left(\frac{0 + 0,1167}{0,0944 + 0,1167} \right)$$

$$= 0,210563713$$

Selanjutnya dilakukan prosedur yang sama untuk persamaan (3.5) dengan menggunakan metode *shooting* dan dengan nilai tebakan $s = 0,21056371$. Kemudian dilakukan perulangan sampai iterasi ke-10 sehingga diperoleh nilai $u(3)$ seperti pada Tabel 10 berikut ini:

Tabel 10 Nilai $u(3)$ dengan Pendekatan Interpolasi Linear

x_n	2,2	2,3	2,4	2,5	2,6	2,7	2,8	2,9	3
$u(x_n)$	0,0325	0,0421	0,0474	0,0487	0,0462	0,0399	0,0301	0,0168	0

Dari Tabel 10 di atas dapat diketahui bahwa pada iterasi ke-10 yaitu $x_{10} = 3$ diperoleh nilai $u(3) = y(3) = 0$. Hal ini menunjukkan bahwa perhitungan secara numerik dari persamaan (3.5) mendekati nilai batas sebenarnya yaitu $y(3) = 0$.

Perhitungan menggunakan metode *shooting* pada permasalahan II dilakukan sebanyak tiga kali, yang mana dua tebakan awal adalah tebakan acak. Kemudian tebakan selanjutnya adalah tebakan yang memanfaatkan dua tebakan awal sehingga didapatkan hasil $y'(2) = 0.21056371$, dan ternyata menghasilkan solusi yang akurat. Hal ini dimungkinkan karena dua tebakan acak awal berdekatan dengan solusi eksaknya. Selanjutnya akan dicari nilai $y'(2)$ menggunakan metode *shooting neural networks*. Sesuai dengan algoritma *shooting neural networks*, langkah awal yang harus dilakukan adalah melakukan metode *shooting* dengan beberapa nilai tebakan ($y'(2)$) kemudian melakukan metode runge kutta orde 4 untuk mendapatkan nilai pada batas akhirnya ($y(3)$). Misalkan diketahui lima nilai tebakan dan nilai dibatas akhirnya seperti pada Tabel 11 berikut:

Tabel 11 Nilai Tebakan dan Nilai Batas pada Permasalahan II

$y'(2)$	-1	0	1	2	3
$y(3)$	-1.2778	-0.2222	0.8333	1.8889	2.9444

Tabel 11 ini merupakan tabel pembelajaran *neural networks* yang nantinya digunakan untuk mengetahui pola dari *neural networks*. Karena dari soal yang diketahui adalah $y(3) = 0$, dan tujuan adalah ingin mengetahui nilai $y'(2)$ maka dalam menginputkan ke *neural networks* Tabel 11 dapat dituliskan menjadi Tabel 12 berikut:

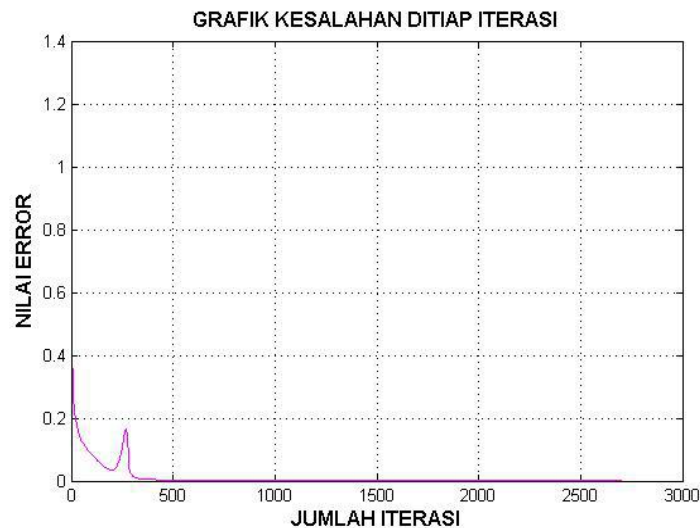
Tabel 12 Nilai Input dan Target pada Permasalahan II

Input ($y(3)$)	-1.2778	-0.2222	0.8333	1.8889	2.9444
Target ($y'(2)$)	-1	0	1	2	3

Sesuai dengan algoritma *shooting neural networks* arsitektur neural networks yang akan digunakan sama seperti pada Permasalahan I. Dengan menggunakan parameter *neural networks* seperti pada Tabel 13 maka didapatkan grafik kesalahan pembelajaran seperti pada Gambar 5.

Tabel 13 Nilai Parameter *Neural Network* pada Permasalahan II

Jumlah node layer 1	Jumlah node output	Learning rate (α)	Kesalahan maksimum
5		1.5	0.0001



Gambar 5 Grafik Kesalahan Pembelajaran Permasalahan 2 di Setiap Iterasinya

Dari grafik dapat dilihat nilai error pada iterasi terakhir adalah kurang dari 0,0001. Artinya diperoleh nilai bobot v dan w yang sudah optimal sehingga ketika diinputkan nilai -1,2778 maka nilai keluaran yang didapat adalah -1,0001. Nilai bobot v dan w tersebut digunakan untuk menentukan nilai $y'(2)$. Dengan menginputkan nilai batas akhir yang sudah diketahui pada permasalahan II yaitu $y(3) = 0$ maka didapatkan nilai $y'(2) = 0.2141$.

Dari ketiga perhitungan di atas, didapatkan hasil pada Tabel 14. Dengan menggunakan nilai syarat tersebut dan metode runge kutta orde 4 diperoleh perbandingan tiga grafik yang dapat dilihat pada Gambar 6.

Tabel 14 Syarat Nilai pada Permasalahan II

$y'(2)$ eksak	$y'(2)$ metode shooting	$y'(2)$ metode shooting neural networks
0,2105	0,21056371	0,2141

Gambar 6 Grafik Perbandingan Tiga Nilai Fungsi dengan Syarat $y'(2)$ Eksak, $y'(2)$ Metode Shooting, $y'(2)$ Metode Shooting Neural Networks

Dari gambar 6 dapat dilihat bahwa grafik *shooting* hampir sama dengan grafik eksaknya. Sedangkan grafik *shooting neural networks* berada dibawah grafik *shooting* dan grafik eksak. Hal ini disebabkan karena nilai $y(2)$ yang digunakan pada grafik *shooting* hampir sama dengan nilai eksaknya sedangkan nilai $y'(2)$ yang digunakan pada *shooting neural networks* nilainya hanya mendekati nilai $y'(2)$ eksak, sehingga $y(3)$ yang didapatkan pada *shooting neural networks* tidak sama dengan $y(3)$ eksak, tetapi hanya mendekati. Dari permasalahan II dapat disimpulkan bahwa metode *shooting* memiliki nilai yang lebih akurat daripada nilai yang didapat dari *shooting neural networks*. Hal ini disebabkan karena tebakan awal nilai *shooting* tepat mengenai solusi eksaknya.

4. KESIMPULAN

Algoritma *Neural Networks* dapat digabungkan dengan Metode *Shooting* untuk mengerjakan permasalahan syarat batas dengan dua titik. Hasil $y'(a)$ dengan $a = \text{titik batas awal}$, yang didapat dengan menggunakan metode *shooting* lebih bagus daripada metode *shooting neural networks* jika nilai tebakan titik pada metode *shooting* semakin mendekati nilai $y'(a)$ eksak. Metode *shooting neural networks* baik digunakan untuk permasalahan yang tidak diketahui solusi eksaknya.

DAFTAR PUSTAKA

- [1] Boyce, W., DiPrima, R. 2001. *Elementary Differential Equation Seven Edition*. John Willey and Sons Inc, New York.
- [2] Chapra, et al. 2002. *Numerical Methods for Engineers*. McGraw-Hills Companies, Michigan.
- [3] Kusumadewi, S. 2004. *Membangun Jaringan Syaraf Tiruan Menggunakan Matlab dan Excel Link*. Graha Ilmu, Yogyakarta.
- [4] Mursaid. 1992. *Dasar-Dasar Analisis Numerik*, Terjemahan. PT. Gelora Aksara Pratama, Jakarta.
- [5] Patterson, D.W. 1996. *Artificial Neural Networks Theory and Applications*. Prentice Hall Simon and Schuster Pte Ltd, Singapore.
- [6] Setiawan, K. 2003. *Paradigma Sistem Cerdas*. Bayumedia Publishing, Malang.
- [7] Siang, J. H. 2005. *Jaringan Syaraf Tiruan & Pemrogramannya Menggunakan Matlab*. Andi, Yogyakarta.