

ANALISIS METODE RBF-NN DENGAN OPTIMASI ALGORITMA GENETIKA PADA PERAMALAN MATA UANG EUR/USD

Nengah Widiangga¹, Agus Dharma², Made Sudarma³

Abstract—This paper discuss about EUR/USD forex forecasting using RBF-NN (Radial Basis Function – Neural Network) method without optimization and RBF-NN optimized by 3-models of Genetic Algorithm (GA) and Adaptive Genetic Algorithm (AGA). Forecasting, which is done with the help of MATLAB program, uses 2 types of data (daily high and daily low) and 2 to 5 clusters model. The accuracy is shown as MAPE (Mean Absolute Percentage Error) value. In daily low data, method with the best MAPE is GA type II (3 clusters) with MAPE value of 0.2286%, while in daily high data, method with the best MAPE is AGA type II (3 clusters) with MAPE value of 0.2190%. GA type II and AGA type II has certain technique which searches around RBF-NN's weight, and this technique proofed to be effective in EUR/USD case. Improvement in accuracy that GA type II and AGA type II gives to RBF-NN method may also be used in other currency as well.

Intisari—Penelitian ini membahas tentang peramalan EUR/USD menggunakan metode RBF-NN (*Radial Basis Function – Neural Network*) tanpa optimasi dan RBF-NN yang dioptimasi dengan 3 model AG/AGA (Algoritma Genetika dan Algoritma Genetika Adaptif). Sistem RBF-NN dapat diterapkan pada data dengan karakteristik nonlinear dan fluktuatif seperti data EUR/USD. Permasalahan akurasi muncul jika terjadi solusi lokal dalam sistem RBF-NN dan metode AG/AGA dapat digunakan untuk mengatasi solusi lokal tersebut. Keakuratan dari peramalan ditunjukkan lewat nilai MAPE (*Mean Absolute Percentage Error*). Pada data *daily low*, metode terbaik adalah Algoritma Genetika II dengan MAPE sebesar 0,2286%, sementara pada data *daily high* metode terbaik adalah Algoritma Genetika Adaptif II dengan MAPE sebesar 0,2190%. Metode AG II dan AGA II didukung teknik pencarian di dekat bobot RBF-NN yang terbukti efektif pada kasus mata uang EUR/USD. Perbaikan akurasi yang diberikan AG II dan AGA II terhadap metode RBF-NN dapat diterapkan pada peramalan mata uang lainnya.

Kata Kunci— EUR/USD, RBF-NN, algoritma genetika, algoritma genetika adaptif, cluster, MAPE, *daily high*, *daily low*.

I. PENDAHULUAN

Pasar mata uang atau sering juga disebut Forex adalah pasar yang terus beraktivitas selama 24 jam per hari, 7 hari per minggu. Salah satu komoditas yang diperjualbelikan di Forex

adalah mata uang Euro (EUR) dan US Dollar (USD).

Kurs atau nilai perbandingan dari kedua mata uang ini disebut dengan EUR/USD sementara orang yang bertransaksi di Forex dengan tujuan mendapatkan keuntungan disebut *Trader*. Data Forex terdiri dari empat bagian yaitu nilai *Open*, *Close*, *Maximum* dan *Minimum* untuk setiap *time frame* yang berbeda (harian, mingguan, 1 jam, 4 jam).

Karakteristik data Forex mirip dengan karakteristik data *financial time series* yang bersifat nonlinear dan fluktuatif. Karena pola yang cenderung rumit, maka sulit untuk melakukan perhitungan manual dengan alat hitung biasa dan melakukan peramalan dengan mengenali pergerakan pola tersebut [1]. Peramalan data Forex adalah proses untuk memperkirakan nilai kurs mata uang tersebut di masa depan. *Trader* memerlukan nilai peramalan untuk menentukan tindakan apa yang akan diambil; membeli, menjual atau menunggu.

Penulis memilih EUR/USD karena mata uang ini didukung oleh dua kekuatan ekonomi terbesar dunia yaitu Amerika Serikat dan Uni Eropa [2]. Sebagai akibatnya EUR/USD bersifat likuid dan berpotensi menghasilkan keuntungan finansial apabila peramalannya memberikan hasil yang akurat.

Berdasarkan jurnal-jurnal yang telah diterbitkan, beberapa peneliti telah mencoba meramalkan data Forex dengan metode Statistik, Fuzzy Logic, Neural Network [3], [4]. Salah satu permasalahan yang ditemui pada peramalan data Forex adalah akurasi yang kurang baik karena data yang bersifat nonlinear dan fluktuatif.

Salah satu model dari Jaringan Syaraf Tiruan (JST) yang dapat digunakan dalam peramalan adalah RBF-NN (*Radial Basis Function – Neural Network*). Jaringan ini menggunakan model yang mentransformasi input menggunakan fungsi *Gaussian* dan dilanjutkan secara linear pada output [5]. Fungsi *Gaussian* bersifat nonlinear sehingga dapat diterapkan untuk data yang memiliki karakteristik nonlinear seperti data Forex. Sistem RBF-NN yang menggunakan LMS (*Least Mean Square*) memiliki keterbatasan akurasi jika dalam pencarian yang terjadi adalah solusi lokal.

Algoritma Genetika (AG) dan Algoritma Genetika Adaptif (AGA) dapat digunakan untuk mengatasi solusi lokal tersebut. AG dan AGA bekerja dengan mencari bobot terbaik pada model RBF-NN [6]. Dalam praktiknya, AG dan AGA memiliki banyak pilihan model tergantung dari jenis seleksi, *crossover* (perkawinan) dan mutasi yang dipilih. Sistem AG dan AGA bergantung pada probabilitas, akibatnya solusi yang terbaik belum tentu bisa diperoleh. Pada penelitian sebelumnya [7], metode RBF-NN dioptimasi dengan 1 model AG dan AGA.

Pada penelitian ini, penulis menganalisis metode RBF-NN dengan optimasi AG dan AGA sebanyak 3 model untuk

¹Mahasiswa Pascasarjana, S2 Teknik Elektro Universitas Udayana, Jln. PB. Sudirman Denpasar Bali 80232 INDONESIA (telp: 0361-239599; e-mail: widiangga@gmail.com)

^{2,3}Dosen Pascasarjana, S2 Teknik Elektro Universitas Udayana, Jln. PB. Sudirman Denpasar Bali 80232 INDONESIA (telp: 0361-239599; e-mail: agus_dharma@unud.ac.id, msudarma@unud.ac.id)



peramalan mata uang EUR/USD. Hasil analisis adalah nilai *error* dalam bentuk MAPE (*Mean Absolute Percentage Error*), semakin kecil MAPE maka metode tersebut semakin baik. Pada setiap *cluster*, mulai dari model dengan 2 hingga 5 *cluster*, ada 7 buah nilai MAPE yang akan dibandingkan yaitu nilai MAPE dari metode RBF-NN tanpa optimasi, 3 model RBF-NN dengan AG dan 3 model RBF-NN dengan AGA. Metode RBF-NN dengan optimasi AG dan AGA bertujuan memberikan hasil peramalan yang lebih baik daripada metode RBF-NN tanpa optimasi.

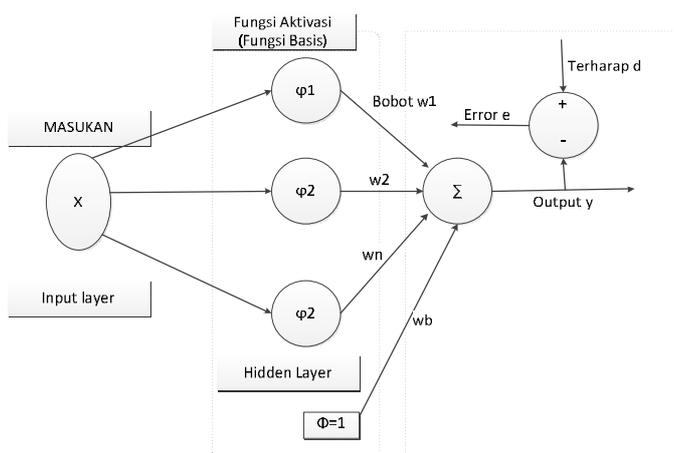
II. KAJIAN PUSTAKA

A. KMEANS Cluster dan RBF-NN

Data *Clustering* merupakan salah satu metode *data mining* yang bersifat tanpa arahan (*unsupervised*). *K-Means* berusaha membagi-bagi data yang ada ke dalam satu atau lebih *cluster*. Metode ini mempartisi data ke dalam *cluster* sehingga data yang memiliki karakteristik yang sama dikelompokkan ke dalam satu *cluster* sementara data dengan karakteristik yang berbeda dikelompokkan ke dalam *cluster* yang lain. Adapun tujuan dari data *clustering* ini adalah untuk meminimalkan *objective function* sehingga meminimalkan variasi di dalam suatu *cluster* dan memaksimalkan variasi antar *cluster*.

Algoritma *K-Means* dimulai dengan pemilihan nilai *K* secara acak, dengan nilai *K* menunjukkan jumlah *cluster* yang ingin dibentuk. Untuk sementara nilai tersebut menjadi pusat dari *cluster* (*centroid*). Selanjutnya perhitungan jarak dilakukan dengan rumus *Euclidian* untuk setiap data yang ada terhadap masing-masing *centroid* hingga didapatkan jarak terdekat. Setiap data lalu diklasifikasikan berdasarkan kedekatannya dengan *centroid*. Langkah-langkah tersebut dilakukan hingga nilai *centroid* tidak berubah (*stabil*).

RBF-NN (φ) merupakan fungsi dengan keluaran yang simetris terhadap *center* (*c*) tertentu atau dinyatakan sebagai $\varphi = \varphi \|x - c\|$, dimana $\| \cdot \|$ merupakan vektor normal. Fungsi yang digunakan misalnya *Gaussian*, *Multiquadric*, *Inverse Quadratic*, *Inverse Multiquadric*, dll. Jaringan RBF terdiri atas tiga *layer* yaitu *layer input*, *hidden layer* dan *layer output*. Struktur dasar jaringan RBF-NN ditunjukkan pada Gambar 1.



Gambar 1: Struktur RBF-NN [7]

Setelah menentukan pusat dan jumlah *neuron*, langkah selanjutnya adalah menentukan matriks φ dan matrik t [8]. Setelah matriks ini ditentukan, langkah selanjutnya adalah mencari bobot (w). Elemen matrik φ dibentuk dengan menggunakan model sebagai berikut: $\left[\exp \left(-\frac{\|data-center\|^2}{2\sigma^2} \right) \right]$; data diambil dari data pengujian atau training dan *center* diambil dari pusat *cluster*. Nilai untuk σ dapat diambil suatu konstanta yang menandakan jarak radial data tersebut dengan *center*.

Matrik t mengambil elemen-elemen dari data pengujian dan training. Persamaan linearnya sesuai dengan (1).

$$\varphi w = t \quad (1)$$

Bobot (w) dapat dicari dengan pendekatan *least-square* sesuai dengan (2).

$$w = (\varphi^T \varphi)^{-1} \varphi^T t \quad (2)$$

Pada penelitian ini, *hidden layer* dari RBF-NN dibentuk menggunakan *center* dari masing-masing *cluster*. Setiap data yang dimasukkan melalui *input layer* akan diolah dalam *hidden layer*, kemudian dikalikan dengan bobot sesuai (1). Hasil perkalian ini merupakan peramalan yang dihasilkan oleh RBF-NN.

B. Algoritma Genetika, Algoritma Genetika Adaptif, MAPE

Algoritma Genetika adalah teknik optimasi dan pencarian berdasarkan prinsip-prinsip genetika dan seleksi alam. Optimasi sendiri dapat diartikan sebagai mencari hasil terbaik dari segala hal yang kita lakukan [9]. Algoritma Genetika dan Algoritma Genetika Adaptif meliputi langkah-langkah yaitu : Representasi Kromosom dan Inialisasi Kromosom, Evaluasi *Fitness*, Seleksi, *Crossover* (perkawinan), Mutasi, *update* probabilitas *crossover* dan probabilitas mutasi (khusus untuk AGA).

Meskipun tidak ada parameter baku untuk setiap kasus yang menggunakan algoritma genetika, penulis menggunakan ketentuan berikut.

- Penggunaan azas elitisme.
- Tingkat *crossover* ditentukan sebesar 70% (berdasarkan peluang *crossover* pada algoritma genetika adaptif).
- Tingkat mutasi ditentukan sebesar 40% (berdasarkan peluang mutasi pada algoritma genetika adaptif).
- Populasi ditentukan sebesar 100.

Tiga model AG yang digunakan dalam penelitian ini adalah sebagai berikut.

- AG Tipe I : *rank-based fitness assignment, single point crossover, single point mutation*. Langkah pertama adalah menginisialisasi kromosom dengan bobot yang sesuai dengan *cluster*, lalu pencarian MAPE dan pemeringkatan (1-100). *Crossover* dilakukan berdasarkan prinsip *single point* dengan bantuan bilangan acak. Bilangan acak ini menentukan titik-titik mana yang akan mengalami *crossover* dan digunakan untuk penambahan tanda minus. Proses mutasi berlangsung dengan mengalikan nilai bobot dengan bilangan acak,

selanjutnya dikalikan kembali dengan 100. Proses AG tipe I akan berakhir jika telah mencapai 50 epoch.

- b. AG Tipe II : *roulette wheel, single point crossover (random number), single point random mutation*. Langkah pertama adalah menginisialisasi kromosom dengan bobot, lalu dilakukan pemeringkatan berdasarkan MAPE. Berikutnya *crossover* dilakukan dengan metode *roulette wheel*; dalam hal ini bobot yang peringkatnya lebih rendah akan digantikan oleh bobot yang peringkatnya lebih tinggi. Proses mutasi melibatkan seluruh kromosom, dimana bobot akan melalui proses pengalihan dengan bilangan acak, bobot minimum dan bobot maksimum. Algoritma Genetika tipe 2 akan berakhir jika telah mencapai 50 epoch, setelah itu MAPE seluruh kromosom akan dihitung.
- c. AG Tipe III : *stochastic selection, single point crossover, mutation with random multiplication*. Langkah pertama adalah menginisialisasi kromosom dengan bobot yang sesuai dengan *cluster*, lalu dilakukan pemeringkatan berdasarkan nilai MAPE. Tahap *crossover* dimulai dengan *stochastic selection* dan dilanjutkan dengan *Single point crossover* yang dilaksanakan 10 kali setiap epoch. Proses mutasi hanya melibatkan kromosom yang mengalami *crossover* melalui mekanisme pengalihan bobot dengan bilangan acak dan perpangkatan acak dari -1 (*mutation with random multiplication*). Algoritma Genetika tipe 3 akan berakhir jika telah mencapai 50 epoch, setelah itu MAPE seluruh kromosom akan dihitung.

AGA pada dasarnya sama dengan AG namun perbedaan mendasar pada AGA adalah adanya penentuan probabilitas *crossover* dan probabilitas mutasi sesuai dengan nilai *fitness*.

Rumus yang digunakan untuk melakukan update probabilitas *crossover* dan probabilitas mutasi sesuai dengan (3) dan (4).

$$p_c = k_1(f_{max} - f')(f_{max} - \hat{f}) \tag{3}$$

$$p_m = k_2(f_{max} - f')(f_{max} - \hat{f}) \tag{4}$$

dengan batasan

$$k_1, k_2, k_3 \leq 1.0$$

Keterangan :

p_c adalah probabilitas *crossover*

p_m adalah probabilitas mutasi

f_{max} adalah *fitness* maksimal

f adalah *fitness* terbesar dari solusi yang dimutasi

f' adalah *fitness* terbesar dari solusi yang disilangkan

\hat{f} adalah rata-rata *fitness*

Nilai $p_m = 0,6$, $p_c = 0,3$, $k_1 = 0,7$ dan $k_2 = 0,3$. Nilai-nilai ini ditetapkan dengan harapan memperbesar peluang untuk *crossover* dari kromosom-kromosom yang memiliki *fitness* baik, dan memperbesar peluang mutasi dari kromosom-kromosom dengan *fitness* jelek [7].

Tiga model AGA yang digunakan dalam penelitian ini adalah sebagai berikut.

- a. AGA Tipe I
Langkah pertama adalah menginisialisasi kromosom lalu pencarian MAPE dan pemeringkatan. Tahap *crossover* dilakukan berdasarkan probabilitas *crossover* pada kromosom 1-6 dan nilai *Update*. *Crossover* berlangsung jika bilangan acak lebih dari atau sama dengan 0,5 dan nilai *Update* lebih besar daripada *probabilitas crossover*. Proses mutasi berlangsung jika nilai *update* lebih besar daripada probabilitas mutasi. Algoritma Genetika Adaptif tipe 1 akan berakhir jika telah mencapai 50 epoch, setelah itu MAPE seluruh kromosom akan dihitung.
- b. AGA Tipe II
Langkah pertama pada Algoritma Genetika Adaptif Tipe II adalah menginisialisasi kromosom lalu pencarian MAPE dan pemeringkatan. Kemudian *crossover* dilakukan dengan metode *roulette wheel*; dalam hal ini bobot yang peringkatnya lebih rendah akan digantikan oleh bobot yang peringkatnya lebih tinggi. *Crossover* berlangsung jika nilai *Update* lebih besar daripada *probabilitas crossover*. Penentuan *crossover* dengan *roulette wheel* dilaksanakan 10 kali setiap epoch. Proses mutasi ditentukan oleh nilai *Update* setiap kromosom. Algoritma Genetika Adaptif tipe 2 akan berakhir jika telah mencapai 50 epoch, setelah itu MAPE seluruh kromosom akan dihitung.
- c. AGA Tipe III
Langkah pertama pada Algoritma Genetika Adaptif Tipe III adalah menginisialisasi kromosom lalu pencarian MAPE dan pemeringkatan. Selanjutnya dilakukan *stochastic selection*, dimana kromosom yang akan melakukan *crossover* ditentukan lewat bilangan acak. *Crossover* berlangsung jika nilai *Update* lebih besar daripada *probabilitas crossover*. Proses mutasi ditentukan oleh nilai *Update* setiap kromosom. Algoritma Genetika Adaptif tipe 3 akan berakhir jika telah mencapai 50 epoch, setelah itu MAPE seluruh kromosom akan dihitung.

C. MAPE (Mean Absolut Percentage Error)

MAPE sesuai dengan (5) digunakan untuk untuk menilai kinerja suatu proses dengan menyatakan perbedaan antara data ramalan (F_t) dengan data aktual (A_t) dalam bilangan riil melalui persentase dengan mengabaikan indikator positif atau negatif pada galat [10].

$$MAPE = \left(\frac{100}{n}\right) \sum_{t=1}^n \left| \frac{A_t - F_t}{A_t} \right| \tag{5}$$

dengan n adalah banyaknya data.

III. METODELOGI PENELITIAN

A. Tahapan Penelitian

Tahapan pada penelitian ini mencakup 3 hal yaitu :



1. *Pre-processing* yang meliputi proses pengambilan data dan membagi data tersebut menjadi data training dan data pengujian.
2. *Processing* yang meliputi pembentukan arsitektur jaringan RBF-NN dan optimasi bobot dengan AG dan AGA.
3. *Post processing* yang meliputi pengujian model RBF-NN dan RBF-NN yang dioptimasi dengan AG/AGA.

B. Pengambilan dan Pengolahan Data

Pengambilan data training dan data pengujian meliputi langkah-langkah berikut.

- a. Membuka terminal MT4 Liteforex.
- b. Pilih *tools*, masuk ke *History Centre* kemudian memilih datanya (dalam hal ini *Daily* untuk kurs EUR/USD) dan klik *Export*. Format *default*-nya adalah dalam bentuk *csv*.
- c. Data dalam format *csv* dibuka dengan MS Excel lalu di-edit untuk mengambil data *daily high* dan *daily low*.
- d. *Macro VBA* digunakan untuk mengambil 1000 data training dan 300 data pengujian. Data ini akan disimpan ke dalam variabel *MATLAB*.

Data training menggunakan data tanggal 20 Oktober 2009 hingga 22 Agustus 2013, sementara data pengujian menggunakan data tanggal 23 Agustus 2013 hingga 17 Oktober 2014.

C. Pembentukan Cluster dan Pemodelan RBF-NN

Pembentukan *cluster* menggunakan 5 buah program. Masing-masing program mewakili jumlah *cluster* yang berbeda (2 – 5 *cluster*). *Clustering* dengan metode *K-means cluster* memanfaatkan fungsi bawaan dari *MATLAB* yaitu *kmeans()*. Nilai *center* yang dihasilkan fungsi *kmeans()* akan dimasukkan ke dalam fungsi basis RBF-NN.

Langkah-langkah pembentukan Matrik ϕ dan t adalah sebagai berikut:

- a. Matrik ϕ memiliki jumlah kolom sesuai dengan jumlah *cluster* dan barisnya merupakan data training urutan pertama hingga 999. Kolom pertama dibentuk berdasarkan pusat *cluster* pertama, kolom kedua berdasarkan pusat *cluster* kedua, dst. Jika data pertama adalah 1,2837 dan data kedua adalah 1,2845, maka pusat *cluster* pertama adalah 1,2500 dan pusat *cluster* kedua adalah 1,2600, maka matrik ϕ yang dibentuk dari 2 data awal ini adalah :

$$b. \left[\exp\left(-\frac{\|1.2837 - 1.2500\|^2}{2\sigma^2}\right) \quad \exp\left(-\frac{\|1.2845 - 1.2600\|^2}{2\sigma^2}\right) \right]$$

dengan σ diambil sama dengan 1, yang merupakan pendekatan fungsi Gaussian dengan radial sebesar 1 terhadap data Forex.

- c. Matrik t terdiri dari data training urutan kedua hingga urutan seribu. Matrik t adalah sasaran yang akan dicapai dengan menyesuaikan nilai bobot. Matrik t yang terbentuk dari poin a di atas adalah : [1,2845]. Jadi untuk 2 *cluster* dan 1000 data training akan

terbentuk matrik ϕ (999 x 2), matrik w (2 x 1) dan matrik t (999 x 1). Pada struktur RBF-NN dengan 3 *cluster*, matrik ϕ yang terbentuk berdimensi (999 x 3), matrik w berdimensi (3 x 1) dan matrik t (999 x 1). Bobot w dapat dihitung melalui (2).

- d. Setelah matrik w ditentukan, maka hasil perkalian matrik ϕ dan matrik w merupakan hasil peramalan atau keluaran dari sistem, sementara matrik t merupakan target yang harus dicapai. Penghitungan *error* (MAPE) dilakukan sesuai (5), jadi hasil perkalian ϕ dan w diambil sebagai nilai A_t , nilai F_t diambil dari matrik t , dan n sesuai jumlah data. Misalkan hasil peramalan ke-1 sampai ke-5 memberikan nilai : 1,1 ; 1,2 ; 1,3 ; 1,4 ; 1,5 dan nilai dari matrik t sebagai target ke-1 sampai ke-5 adalah : 1,2 ; 1,2 ; 1,3 ; 1,2 ; 1,5, maka nilai MAPE yang diperoleh :

$$MAPE = \left(\frac{100}{5}\right) \left[\left| \frac{1,2-1,1}{1,2} \right| + \left| \frac{1,2-1,2}{1,2} \right| + \left| \frac{1,3-1,3}{1,3} \right| + \left| \frac{1,2-1,4}{1,2} \right| + \dots + \left| \frac{1,5-1,5}{1,5} \right| \right] = 0,2778.$$

- e. Proses pada *cluster* lainnya (selain 2 *cluster*) berjalan dalam tahapan yang sama. Perubahan hanya terjadi pada dimensi matrik ϕ dan matrik w saja.

D. Optimasi dengan AG/AGA

Setiap elemen dari matrik w akan menjadi kromosom pada proses optimasi, diikuti dengan proses penambahan kromosom agar menjadi 100 buah menggunakan bantuan bilangan acak. Kromosom ini akan mendapat perlakuan yang berbeda-beda tergantung tipe AG dan AGA yang digunakan.

Misalnya pada AG tipe I dengan 2 *cluster*, matrik w yang terbentuk adalah sebagai berikut:

$$\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & \dots & 100 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & \dots & 100 \end{bmatrix}$$

Jika *crossover* dengan bilangan acak memenuhi ketentuan bahwa bobot 91 hingga 100 diganti dengan 1 sampai dengan 10, maka elemen di kolom 91 hingga 100 akan diganti dengan elemen pada kolom 1 hingga 10. Jika bilangan acak yang dihasilkan memberikan ketentuan bahwa elemen 91 hingga 100 diberikan tanda negatif, maka matrik w yang baru terbentuk akan menjadi :

$$\begin{bmatrix} 1 & \dots & 1,1 & 12 & 0,3 & 1,5 & 5,3 & 4 & 1,7 & 2,1 & -2 & 3 \\ 1 & \dots & 1,1 & 12 & 0,3 & 1,5 & 5,3 & 4 & 1,7 & 2,1 & -2 & 3 \end{bmatrix}$$

Proses mutasi dilakukan dengan mengalikan bobot pada kolom 91 hingga 100 dengan bilangan acak lalu dikalikan kembali dengan 100. Jika syarat untuk mutasi terpenuhi maka matrik w yang terbentuk adalah :

$$\begin{bmatrix} 1 & \dots & 0,23 & 2,2 & 3,12 & 4,87 & 1,5 & 2,6 & 0,73 & 1,432 & -1,201 & 0,5 \\ 1 & \dots & 0,18 & 2,4 & 3,14 & 4,85 & 1,57 & 2,6 & 0,72 & 1,432 & -1,201 & 0,5 \end{bmatrix}$$

Setelah matrik w diperoleh, maka proses selanjutnya adalah memasukkan kembali matrik w ini ke dalam skema RBF-NN

di atas, untuk mencari MAPE-nya. Demikian seterusnya hingga proses mencapai batas akhir yaitu 50 epoch.

Proses AGA memiliki sedikit perbedaan pada tahapan *crossover* dan mutasi. Jika diambil matrik *w* pada contoh sebelumnya :

$$\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & \dots & 100 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & \dots & 100 \end{bmatrix}$$

maka harus ditentukan dulu nilai *fitness* dari masing-masing kromosom, dengan cara mencari nilai MAPE-nya. Untuk mencari MAPE, caranya sama dengan pada metode AG. Nilai *fitness* lalu ditentukan berdasarkan perbandingan terbalik dari MAPE.

Jadi jika diasumsikan hanya ada 4 kromosom, sementara nilai MAPE yang diperoleh kromosom 1 adalah 0,1; kromosom 2 adalah 0,2; kromosom 3 adalah 0,3; kromosom 4 adalah 0,4; maka kromosom 1 memiliki proporsi 10, kromosom 2 memiliki proporsi 5, kromosom 3 sebesar 3,3 dan kromosom 4 sebesar 2,5. Nilai total proporsi menjadi 20,8. *Fitness* dari kromosom 1 menjadi 0,487; kromosom 2 menjadi 0,24; kromosom 3 menjadi 0,144; dan kromosom 4 menjadi 0,12. Nilai *fitness* ini akan digunakan dalam perhitungan untuk menentukan apakah kromosom itu layak untuk menjalani proses *crossover* dan mutasi atau tidak. Jika misalnya ditentukan $p_c = 0,3$, dengan $k_1 = 0,7$ maka perhitungan probabilitas *crossover* untuk kromosom 2 menjadi :

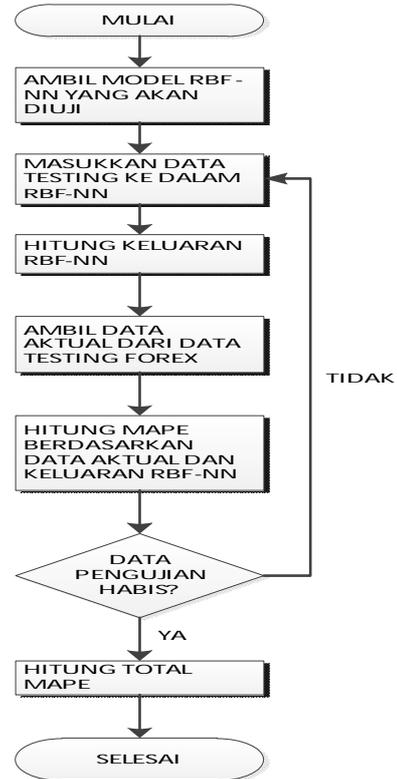
$$p_2 = 0,7 \times (0,487 - 0,24) / (0,487 - 0,248) = 0,041.$$

Karena nilai probabilitasnya yang tidak sama dengan atau lebih dari p_c maka kromosom 2 tidak menjalani proses *crossover*. Perhitungan untuk probabilitas mutasi dilakukan dengan cara yang sama.

E. Proses Pengujian

Proses pengujian menggunakan matrik *w* dan ϕ yang dihasilkan dari proses sebelumnya dan matrik *t* yang berisikan data pengujian. Penghitungan *error* (MAPE) dilakukan melalui cara yang sama dengan perhitungan RBF-NN. Hal ini terus dilakukan sampai data pengujian habis, seperti ditunjukkan pada Gambar 2.

Seperti diutarakan sebelumnya pada Pendahuluan, penulis menggunakan model RBF-NN dengan 2 *neuron* hingga 5 *neuron* (sesuai jumlah *cluster*). Pada setiap *neuron*, akan dihasilkan 1 nilai MAPE dari metode RBF-NN murni dan 6 nilai MAPE dari metode RBF-NN yang telah dioptimasi. Perbandingan dilakukan untuk seluruh nilai MAPE dan metode peramalan dengan nilai MAPE terkecil dianggap sebagai yang terbaik.



Gambar 2: Algoritma pengujian model RBF-NN

IV. HASIL DAN ANALISIS

Berdasarkan percobaan yang telah dilakukan, rangkuman MAPE untuk data *daily low* diberikan pada Tabel I dan untuk data *daily high* diberikan pada Tabel II.

TABEL I
RANGKUMAN MAPE DATA DAILY LOW

CLUSTER	METODE	MINIMUM	MAKSIMUM	RATA-RATA	RANGE
2	RBF	0,2850	0,2850	0,2850	0,0000
	AG I	3,8148	892,8504	247,9353	889,0356
	AG II	0,2564	0,2735	0,2661	0,0171
	AG III	1,6456	371,4353	80,4266	369,7897
	AGA I	1,6089	22,7953	5,7681	21,1864
	AGA II	0,2394	0,2850	0,2656	0,0456
3	AGA III	1,6086	8,1860	2,9226	6,5773
	RBF	0,2297	0,2297	0,2297	0,0000
	AG I	1,6276	319,1932	157,3949	317,5655
	AG II	0,2286	0,2297	0,2290	0,0011
	AG III	1,6402	4249,8330	948,1636	4248,1929
	AGA I	2,5808	18,6688	8,2317	16,0880
4	AGA II	0,2287	0,2295	0,2289	0,0008
	AGA III	1,5064	10,9515	4,7124	9,4451
	RBF	0,2297	0,2301	0,2299	0,0004
	AG I	13,7706	768,4714	248,1263	754,7007
	AG II	0,2287	0,2292	0,2289	0,0005
	AG III	5,7386	401,1650	99,8861	395,4264
5	AGA I	2,9206	30,2798	12,7766	27,3592
	AGA II	0,2286	0,2292	0,2289	0,0006
	AGA III	1,7844	6,6661	3,7177	4,8816
	RBF	0,2296	0,2314	0,2305	0,0018
	AG I	2,5824	388,7684	161,0236	386,1860
	AG II	0,2289	0,2295	0,2292	0,0005
5	AG III	7,7424	4070,4841	557,4044	4062,7417
	AGA I	2,7496	20,3761	8,6517	17,6265
	AGA II	0,2287	0,2289	0,2288	0,0002
	AGA III	1,7087	10,2382	4,8698	8,5295



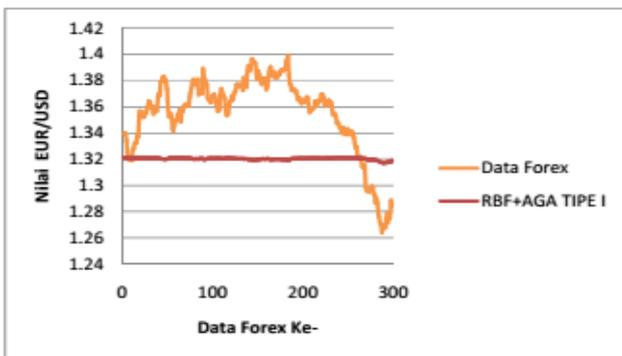
TABEL III
RANGKUMAN MAPE DATA DAILY HIGH

CLUSTER	METODE	MINIMUM	MAKSIMUM	RATA-RATA	RANGE
2	RBF	0,2922	0,2977	0,2950	0,0055
	AG I	15,5548	245,6913	156,6447	230,1365
	AG II	0,2339	0,2874	0,2766	0,0534
	AG III	17,7578	522,5397	126,9847	504,7819
	AGA I	1,5494	28,7295	8,7868	27,1801
	AGA II	0,2576	0,2983	0,2814	0,0407
	AGA III	1,6291	5,3581	2,6282	3,7290
3	RBF	0,2195	0,2195	0,2195	0,0000
	AG I	6,8694	360,1166	116,9075	353,2472
	AG II	0,2192	0,2200	0,2194	0,0008
	AG III	3,5786	718,5100	142,8801	714,9315
	AGA I	1,6860	15,0499	7,9187	13,3640
	AGA II	0,2190	0,2197	0,2193	0,0007
	AGA III	1,6927	7,9883	4,1912	6,2956
4	RBF	0,2196	0,2198	0,2197	0,0002
	AG I	92,8449	344,3286	198,9158	251,4837
	AG II	0,2193	0,2201	0,2198	0,0007
	AG III	3,7275	4459,4744	638,5886	4455,7470
	AGA I	1,6218	31,7179	10,4880	30,0961
	AGA II	0,2193	0,2199	0,2197	0,0006
	AGA III	1,4371	11,6124	4,0321	10,1753
5	RBF	0,2195	0,2204	0,2200	0,0009
	AG I	134,8533	739,3607	340,8346	604,5074
	AG II	0,2193	0,2201	0,2197	0,0008
	AG III	7,1666	761,6106	154,6047	754,4440
	AGA I	3,0750	19,7273	8,0105	16,6523
	AGA II	0,2194	0,2201	0,2198	0,0007
	AGA III	1,6338	13,7531	5,1754	12,1193

Nilai MAPE yang diperoleh pada seluruh *cluster*, seperti terlihat pada Tabel I dan Tabel II, menunjukkan bahwa metode AG II dan AGA II mampu memberikan optimasi terhadap hasil peramalan oleh RBF-NN.

Perbaikan pada RBF-NN tidak memberikan optimasi yang signifikan pada model dengan banyak *cluster* [7]. Hal tersebut sejalan dengan penelitian ini dimana pada model dengan 3 hingga 5 *cluster*, selisih MAPE antara RBF-NN tanpa optimasi dengan RBF-NN yang dioptimasi AG II dan AGA II tidak sebesar pada model dengan 2 *cluster*.

Metode selain AG II dan AGA II memberikan kurva hasil peramalan yang bentuknya mendatar, tidak variatif dan tidak mengikuti data Forex aktual seperti ditunjukkan Gambar 3.

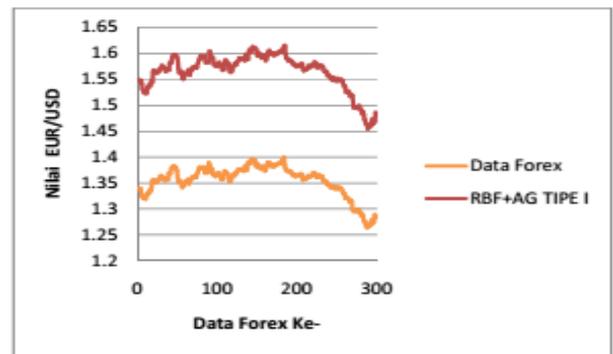


Gambar 3 : Kurva hasil peramalan RBF-NN + AGA I yang nyaris mendatar

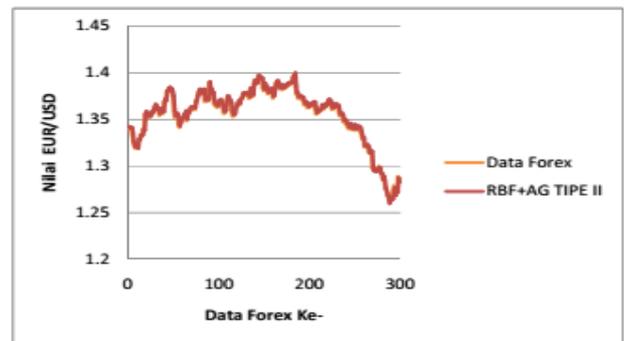
Kurva yang mendatar ini disebabkan oleh nilai bobot yang kecil, yang berpengaruh terhadap hasil peramalan, dimana hasil peramalan nilainya ditentukan berdasarkan perkalian bobot dengan fungsi ϕ .

Selain itu, hasil peramalan dari metode selain AG II dan AGA II memiliki *initial point* yang jauh, sehingga meskipun kurva peramalan bentuknya mirip namun nilai MAPE-nya tetap besar, seperti ditunjukkan pada Gambar 4. *Initial point* yang jauh adalah konsekuensi dari pencarian secara acak di dalam *search space* (ruang pencarian) yang baru.

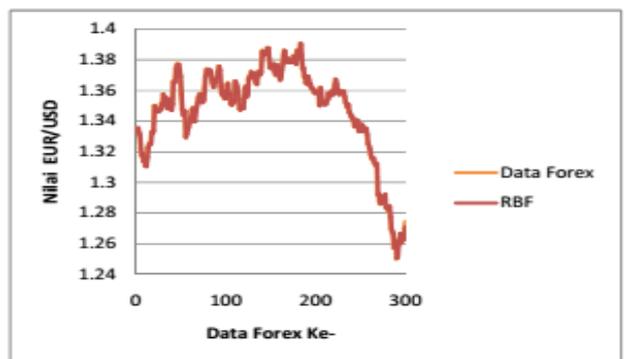
Kurva hasil peramalan AG II ditunjukkan pada Gambar 5, dan kurva hasil peramalan RBF-NN tanpa optimasi ditunjukkan pada Gambar 6. Seperti terlihat pada Gambar 6, kurva hasil peramalan berhimpit dengan kurva data forex aktual.



Gambar 4 : Kurva hasil peramalan RBF-NN + AG Tipe I dengan *initial point* terlalu jauh



Gambar 5 : Kurva hasil peramalan RBF-NN + AG II



Gambar 6 : Kurva hasil peramalan RBF-NN tanpa optimasi

Keunggulan Algoritma Genetika II dan Algoritma Genetika Adaptif II didukung oleh langkah pencarian yang dilakukan di dekat nilai bobot yang dihasilkan RBF-NN tanpa optimasi [7]. Hal ini terlihat dari *range* MAPE dari AG II dan AGA II yang kecil pada *cluster* 3, 4, 5 dan menyerupai *range* MAPE milik RBF-NN tanpa optimasi.

Langkah pencarian di dekat bobot RBF-NN ini terbukti lebih efektif dalam kasus mata uang EUR/USD daripada melakukan pencarian di *search space* (ruang pencarian) yang baru, seperti yang dilakukan metode AG/AGA selain AG II dan AGA II.

V. KESIMPULAN DAN SARAN

Kesimpulan yang diperoleh dari penelitian ini adalah sebagai berikut.

1. Pada data *daily low*, metode terbaik adalah Algoritma Genetika II (3 *cluster*) dengan MAPE sebesar 0,2286%, sementara pada data *daily high* metode terbaik adalah Algoritma Genetika Adaptif II (3 *cluster*) dengan MAPE sebesar 0,2190%. Metode AG II dan AGA II mampu memberikan optimasi yang ditunjukkan dengan nilai MAPE-nya yang lebih kecil daripada metode RBF-NN tanpa optimasi. Nilai MAPE terbaik yang dihasilkan AG/AGA cenderung fluktuatif karena metode ini didasarkan pada mekanisme probabilitas.
2. Metode AG II dan AGA II memberikan kurva hasil peramalan yang mendekati data Forex aktual karena didukung teknik pencarian di dekat bobot RBF-NN. Metode AG/AGA yang lain memberikan MAPE yang besar karena kurva hasil peramalan yang bentuknya cenderung mendatar, tidak variatif dan *initial point* jauh dari data Forex aktual.

Satu kali simulasi RBF-NN dengan optimasi AG/AGA dalam MATLAB dapat memakan waktu hingga 15 menit. Penelitian selanjutnya dapat mencoba cara/teknik lain sehingga waktu simulasi bisa dipersingkat.

REFERENSI

- [1] Huda, F. A., Ridok. A., Dewi. C., 2013, Peramalan Time Series Saham Menggunakan Backpropagation Neural Network Berbasis Algoritma Genetika, Teknik Informatika PTIIIK, Universitas Brawijaya.
- [2] Bank for International Settlements, 2014. Triennial Central Bank Survey. Monetary and Economic Department.
- [3] Georgios Sermpinis, Konstantinos Theofilatos, Andreas Karathanasopoulos, Efstratios F. Georgopoulos, Christian Dunis, 2012. Forecasting Foreign Exchange Rates with Adaptive Neural Networks Using Radial-Basis Functions and Particle Swarm Optimization. European Journal Operation Research 225, 528–540. doi:http://dx.doi.org/10.1016/j.ejor.2012.10.020.
- [4] Zuliana, S.U., 2012. Penerapan Global Ridge-Regression Pada Peramalan Data Time Series Non Linear Studi Kasus : Pemodelan Nilai Tukar US Dollar Terhadap Rupiah. Kaunia VIII.
- [5] Warda, S. M., Irhamni F., 2012. Analisa Data Antaran Pos Express Menggunakan Metode Radial Basis Function Neural Network (RBF-NN) Study PT. POS Indonesia (Persero) Surabaya. Universitas Trunojoyo.
- [6] Maillard, E.P., Gueriot, D., 1997. RBF neural network, basis functions and genetic algorithm, in: International Conference on Neural Networks, 1997. Presented at the International Conference on Neural Networks, 1997, pp. 2187–2192 vol.4. doi:10.1109/ICNN.1997.614247.

- [7] Wiharto, Y.S. Palgunadi, Muh Aziz Nugroho, 2013. Analisis Penggunaan Algoritma Genetika Untuk Perbaikan Jaringan Syaraf Tiruan Radial Basis Function. Seminar Nasional Teknologi Informasi Dan Komunikasi 2013. SENTIKA 2013.
- [8] Heriyanto, D. N., Rachman, F.H., Satoto, B.D., 2013. Penerapan Metode Radial Basis Function Network dengan K-Means Cluster untuk Peramalan Kebutuhan Straw. Jurnal Sarjana Teknik Informatika. Universitas Trunojoyo Madura 1, 1–10.
- [9] Andreasson, N., Evgrafov, A., Patriksson, M., 2006. Introduction to Continuous Optimization: Foundations and Fundamental Algorithms, 1st edition. Studentlitteratur AB.
- [10] Dharma, A., Robandi, I., Purnomo, M.H., 2011. Application of Interval Type-2 Fuzzy Logic System in Short Term Load Forecasting on Special Days. IPTEK Journal for Technology and Science 22. doi:10.12962/j20882033.v22i2.65.



[Halaman Ini sengaja di kosongkan]