

Simulasi Sistem Prediksi Arah Gerak Indeks Harga Saham Gabungan Berbasis Adaptive Neuro-Fuzzy Interference System

Joni Fat¹

ABSTRACT: The simulation system using Fuzzy Inference System (FIS) method in this research can predict the daily closing price trend. The simulation system uses technical indicator as input variables. The FIS type is Sugeno. The defuzzification method is wtaver. Neural Networks (NN) was used to update the range of membership function for labels in all variables. The NN was used Gauss method. The simulation output is a price change variable. All the variables used in this system were technical variables, dan all the fundamental variables were ignored. This simulation used four types of simulator. Each simulator would simulate only one moment, except for the 'Total'. The 'Total' simulator would simulate all moments. The moments were bearish, sideways and bullish. The simulation of each moment was intended to know whether the entire system could make a prediction without suspected as just a trend follower. This system accuracy was about 61.55%.

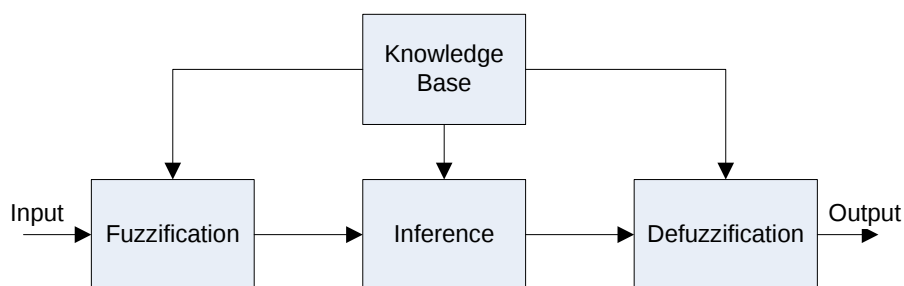
KEYWORDS: fuzzy inference system, neural networks, simulation system, sugeno, technical indicator.

ABSTRAK: Sistem simulasi dengan menggunakan metode Fuzzy Inference System (FIS) dalam penelitian ini memungkinkan prediksi arah gerak harga penutupan saham secara harian. Sistem simulasi menggunakan indikator teknikal sebagai variabel *input*. Tipe FIS yang digunakan adalah Sugeno. Metode defuzzifikasinya adalah *wtaver*. Neural Networks (NN) digunakan untuk menyesuaikan *range* fungsi keanggotaan pada masing-masing label variabel *input* mau pun *output*. NN menerapkan metode Gauss. Hasil simulasi adalah berupa variabel perubahan harga pada periode perhitungan. Variabel-variabel yang digunakan merupakan variabel teknikal, dan meniadakan unsur fundamental. Simulasi ini menggunakan empat buah simulator. Setiap simulator ini mensimulasikan momen-momen tertentu, yaitu momen *bearish*, *sideways*, *bullish* dan total. Momen total merupakan gabungan dari ketiga momen yang lain. Simulasi pada tiap momen berupaya mengetahui apakah sistem hanya mampu melakukan prediksi dengan kecenderungan tertentu saja atau tidak. Bila terbukti, maka simulasi dianggap gagal. Sebaliknya, bila tidak terbukti, maka simulasi dianggap berhasil. Hasil pengujian memperlihatkan simulasi berhasil dengan tingkat akurasi sistem mencapai 61,55%.

KATA KUNCI: fuzzy inference system, indikator teknikal, neural networks, sistem simulasi, sugeno.

PENDAHULUAN

Sistem inferensi *Fuzzy Logic* pada prinsipnya terdiri dari beberapa proses, yaitu fuzzifikasi, inferensi dan defuzzifikasi (perhatikan Gambar 1). Ketiga proses ini didasarkan pada aturan-aturan yang telah dipahami dengan baik. Hanya sistem ini memiliki kelemahan karena tidak memiliki mekanisme untuk mengubah aturan-aturan yang ada sehingga dapat menghasilkan aturan-aturan yang lebih baik [1]. Pada poin inilah, di mana *neural networks* (NN) dapat berperan. Dengan memanfaatkan mekanisme dalam NN, sistem dapat dilatih dan diubah untuk menghasilkan parameter-parameter yang lebih baik.



■ Gambar 1. Blok Diagram Sistem Inferensi Fuzzy

Tetapi NN memiliki kelemahan, yaitu label-label dan aturan-aturan yang dihasilkan menjadi tidak dijelaskan dengan baik. Ini disebabkan label-label dan aturan-aturan tersebut menjadi tidak deskriptif. Hal lain dalam pemanfaatan NN adalah diperlukannya basis data yang mencukupi baik dalam tahap pelatihan mau pun pengujian sistem.

Kelemahan pertama menjadi dapat dimaklumi karena sistem yang dihasilkan dapat dilatih dan diubah untuk menghasilkan parameter-parameter yang lebih baik. Dengan demikian, walaupun sistem menjadi tidak informatif tetapi diharapkan kinerjanya dapat meningkat. Untuk kelemahan kedua, ini dapat diatasi karena data-data yang dimanfaatkan merupakan data nilai Indeks Harga Saham Gabungan (IHSG) yang terbuka untuk publik dan cukup detil ketersediaannya (dengan beberapa pilihan nilai penutupan, yaitu berbasis harian, mingguan, bulanan atau tahunan). Oleh sebab itu, perancangan simulasi sistem untuk prediksi arah gerak harga saham dengan memanfaatkan perpaduan antara *fuzzy logic* NN diharapkan dapat memberikan *output* yang lebih baik dibandingkan dua penelitian terdahulu (tingkat keakuratan simulasi sistem > 56,83% [2], > 60,75% [3]).

Desain dan rancangan simulasi sistem ini diharapkan dapat memberikan prediksi arah pergerakan harga saham dengan keakuratan lebih besar atau sama dengan 60,75% pada periode yang pengujian yang telah ditentukan (momen *bearish*, *sideways* dan *bullish*). Dengan demikian, simulasi ini dapat menjadi acuan dasar

¹ Jurusan Teknik Elektro Universitas Tarumanagara Jakarta

bagi investor ritel dalam mengambil keputusan investasi awal, tanpa perlu terlalu terlibat dengan berbagai masalah teknis lainnya.

Dalam mendesain sistem pengambil keputusan berdasarkan *Fuzzy Logic*, pemahaman sistem adalah salah satu persyaratan dasar. Untuk sistem yang dirancang dalam penelitian ini, sistem belum sepenuhnya dipahami dengan baik. Pemahaman yang dimaksudkan di sini adalah dalam kaitannya dengan proses dan hasil. Pada prinsipnya, penjelasan tentang mekanisme perilaku pasar telah dijelaskan dengan baik oleh teori seperti *Efficient Market Hypothesis* (EMH) atau pun melalui *Behavioral Finance*.

Namun, penjelasan yang ada ini tidak mencakup tentang bagaimana tepatnya memprediksi arah gerak pasar (dalam hal ini diwakili oleh nilai IHSG). Jadi, teori-teori ini hanya menjelaskan bagaimana pasar ini berperilaku dalam kondisi-kondisi yang disyaratkan oleh masing-masing teori tersebut. Oleh sebab itu, penggunaan *neural networks* sebagai pendukung dalam desain sistem ini menjadi sebuah alternatif yang menarik.

Simulasi sistem ini dirancang menggunakan masukan berupa indikator-indikator teknikal yang digunakan oleh para praktisi, yaitu berupa indikator *trend followers* dan *counter-trend* [4]. Hasil simulasi sistem dibatasi berupa indikator untuk menunjukkan arah gerak harga saham (naik atau turun) pada periode yang ditentukan. Sistem menggunakan *software* komputasi teknik, dengan spesifikasi berupa terintegrasi dengan *software* komputasi teknik, memiliki enam buah variabel *input*, dan satu buah variabel *output*.

FUZZY LOGIC

Zadeh menyebutkan bahwa *fuzzy logic* merupakan hal yang sama dengan komputasi dengan perkataan [5]. Dengan menggunakan metode ini, kata atau label digunakan sebagai pengganti angka dalam perhitungan. Demikian diharapkan apa bila ada ketidakakuratan dalam data, dapat ditolerir. Jadi, *fuzzy logic* memungkinkan timbulnya pemahaman yang lebih luas dari pada sekedar benar/salah atau iya/tidak. Walau demikian, hal ini tidaklah berarti *fuzzy logic* sendiri adalah kabur/fuzzy. Sebaliknya, *fuzzy logic* merupakan logika yang akurat untuk ketidakakuratan [6]. Dalam komputasi dengan menggunakan *fuzzy logic*, ada beberapa tahap yang perlu dilakukan.

Tahap pertama adalah mengubah domain asal menjadi domain *fuzzy*. Tahap ini dinamakan fuzzifikasi. Domain asal di sini dimaksudkan sebagai domain non-*fuzzy*. Proses fuzzifikasi ini akan memberikan nilai kepada nilai dari domain asal ini sehingga nilai asal yang sebelumnya dianggap kabur atau tidak jelas ini memiliki kejelasan untuk dapat dikomputasi. Proses ini melibatkan bentuk-bentuk geometris mau pun himpunan diskrit. Bentuk geometris memungkinkan pemberian nilai berlangsung secara kontinu, sedangkan himpunan diskrit seperti tercantum pada namanya, hanya memungkinkan pemberian nilai diskrit. Jadi, secara ringkas fuzzifikasi merupakan suatu proses pemberian nilai.

Tahap kedua adalah inferensi. Tahap ini secara matematis melibatkan proses komposisi terhadap himpunan mau pun relasi *fuzzy*. Proses inferensi melibatkan sekumpulan aturan-aturan yang terdefinisi sehingga setiap aturan akan menghasilkan suatu himpunan atau relasi baru sebagai sebuah kesimpulan. Kumpulan himpunan atau relasi baru ini selanjutnya akan digabungkan melalui proses yang dinamakan agregasi. Melalui proses agregasi ini, suatu kesimpulan *fuzzy* akan dihasilkan.

Tahap ketiga adalah proses defuzzifikasi. Proses ini mengubah kesimpulan *fuzzy* menjadi sebuah kesimpulan yang dapat dipahami dalam domain asalnya. Proses defuzzifikasi ini dapat dilakukan dengan berbagai macam metode, seperti metode *last of maxima*, *first of maxima*, *centroid*, *mean of maxima*, dan lain sebagainya. Pemilihan metode defuzzifikasi dilakukan berdasarkan pengamatan terhadap hasil yang paling memuaskan, tanpa ada suatu panduan baku.

Dengan memahami ketiga tahap dalam proses penarikan kesimpulan dengan menggunakan *fuzzy logic* ini, dapat dipahami bahwa keseluruhan proses ini memerlukan pahaman yang baik terhadap perilaku sistem. Di sinilah terletak kekurangan sekaligus kelebihan dari sistem inferensi ini. Kekurangannya adalah apa bila sistem tidak terpahami dengan baik, maka proses pengembangan sistem akan memerlukan teknik lain untuk mengisi celah tersebut. Kelebihannya adalah apa bila sistem terpahami dengan baik, pengembangan proses inferensi ini dapat dilakukan dengan baik dan memberikan hasil yang memuaskan. Ini dikarenakan proses inferensi menggunakan *fuzzy logic* memungkinkan duplikasi terhadap proses inferensi manusia hampir secara langsung.

Pada perkembangan berikutnya, *fuzzy logic* dipadukan dengan berbagai metode cerdas untuk memudahkan pengembangan sistem. Metode cerdas yang digunakan diantaranya adalah *neural networks* dan algoritma genetik [7]. Oleh sebab itu, *fuzzy logic* dengan metode cerdas banyak diaplikasikan dalam bidang pengontrolan, pengambilan keputusan, riset dan sebagainya.

NEURAL NETWORK

Neural Networks (NN) atau *Artificial Neural Networks* (ANN) merupakan model matematis yang berupaya meniru kemampuan memproses informasi sistem saraf biologis. Model pertama ANN diperkenalkan pertama sekali oleh Warren McCulloch dan Walter Pitts pada tahun 1943. Model ANN ini kemudian mengalami

berbagai macam perbaikan dan pengembangan hingga sekarang. ANN, dalam hal ini fungsi pemetaannya terdiri atas minimum satu lapisan. Lapisan ini dikatakan sebagai lapisan tersembunyi karena lapisan pemetaan ini tidak memerlukan penjelasan lebih lanjut mengenai proses kerjanya. Hal yang menjadi tantangan dalam ANN adalah jumlah lapisan yang terlibat, bagaimana hubungan antar-komponen dalam lapisan dan bagaimana pembobotannya. Di sinilah berperan metode dalam proses pembelajaran ANN.

Ada tiga metode pembelajaran utama ANN, yaitu pembelajaran disupervisi, pembelajaran tanpa supervisi dan pembelajaran dengan penguatan. Dua metode pertama memerlukan data awal sebagai media pembelajaran. Perbedaan kedua ini adalah metode disupervisi berupaya menemukan fungsi yang dapat memetakan seperti yang disyaratkan oleh data, sedangkan metode tanpa supervisi berupaya meminimalkan biaya dengan menggunakan satu jenis pemetaan. Sebaliknya metode penguatan tidak memerlukan data awal yang masif sebagai media pembelajaran, tetapi metode ini memungkinkan sistem mengembangkan diri melalui pembelajaran yang dinamis.

BEHAVIORAL FINANCE

Dalam ilmu Ekonomi Keuangan, diketahui ada dua kelompok besar. Kelompok pertama yang mendukung ketiadaan kemungkinan untuk mengeksploitasi harga sekuritas, yang walau pun dalam perkembangannya mengalami beberapa penyesuaian. Kelompok pertama ini adalah para pendukung *Efficient Market Hypothesis* (EMH). Sebaliknya, kelompok kedua yang mendukung adanya kemungkinan untuk mengeksploitasi harga sekuritas. Kelompok ini adalah para pendukung *Behavioral Finance*. Tetapi pada prinsipnya walau pun dalam EMH, para pendukungnya juga kesulitan menjelaskan berbagai hal tentang harga sekuritas yang dalam beberapa hal memperlihatkan adanya pengeksploitasian. Ini dijelaskan oleh EMH dengan membagi keadaan idealnya menjadi tiga bentuk yang lebih memperlihatkan kondisi nyata, yaitu *weak*, *semi-strong* dan *strong*. Sebaliknya, *behavioral finance* menekankan studi dari pengaruh psikologi terhadap perilaku para praktisi di bidang keuangan dan dampaknya terhadap pasar [8].

Behavioral finance menjelaskan kenapa dan bagaimana pasar tersebut menjadi tidak efisien. Dasar pemikirannya adalah bahwa manusia memiliki kecenderungan berpikir dan bertindak irasional. Irasional ini dapat dibagi menjadi dua kategori, yaitu individu tidak selalu memproses informasi dengan benar dan walaupun memperoleh informasi yang benar, individu dapat membuat keputusan yang salah (bias). Berikut adalah beberapa penyebab kemungkinan terjadinya kesalahan-kesalahan tersebut, yaitu [9]: kesalahan memproses informasi, kesalahan prediksi, terlalu percaya diri, dan konservatisme.

DATA PELATIHAN

Data pelatihan yang digunakan dalam perancangan ini merupakan data harga dan volume perdagangan (dalam satuan lembar) saat penutupan secara harian. Data-data ini di-*download* dari situs *finance.yahoo.com*, dengan kode $\wedge$ JKSE. Kode $\wedge$ JKSE ini merupakan kode untuk Indeks Harga Saham Gabungan (IHSG). Periode yang di-*download* adalah dari Januari 2008 hingga Oktober 2009. Periode ini dipilih karena berdasarkan pengamatan secara visual terhadap grafik data (perhatikan Gambar 2), terlihat bahwa dalam periode ini, ketiga momen (*Bearish*, *Sideways* dan *Bullish*) termuat.

Dari Gambar 2, terlihat bahwa momen *Bearish* (garis merah) berlangsung mulai dari bulan Januari hingga Oktober 2008. Momen *Sideways* (garis biru) mulai dari bulan November 2008 hingga Maret 2009, sedangkan momen *Bullish* (garis hijau) mulai dari bulan April hingga Oktober 2009. Berdasarkan hal ini, didapatkan bahwa untuk momen *Bearish*, jumlah data adalah sebanyak 200 entri. Momen *Sideways* sebanyak 99 entri, sedangkan momen *Bullish* sebanyak 143 entri. Keseluruhan entri data adalah sebanyak 442 entri. Data-data ini akan digunakan sebagai data *training* dan data *checking* simulator.



■ **Gambar 2.** Periode Data Pelatihan dengan Tiga Momen (*Bearish*, *Sideways* dan *Bullish*)

DATA PENGECEKAN

Data pengecekan yang digunakan merupakan data IHSG yang dipilih juga memuat ketiga momen. Dari hasil pengamatan secara visual pada grafik $\Delta JKSE$, didapatkan periode data dari bulan November 2011 hingga bulan Oktober 2012. Jumlah data yang diperoleh data adalah sebanyak 156 entri. Data pengujian ini tidak dilakukan pembedaan jenis momen, tetapi sebaliknya keseluruhan data hanya diperlakukan sebagai entri bagi sistem.

DATA PENGUJIAN

Data pengujian dirancang dalam dua model. Model data rancangan pertama adalah rancangan yang sama dengan rancangan data pelatihan, kecuali tanpa disertai variabel *output %dP*. Model pertama diperlukan untuk mengetahui seberapa efektif sistem setelah dilatih dengan metode *Neural Network* (NN). Model kedua diperlukan untuk benar-benar mengetahui keefektifan sistem hasil rancangan. Apabila pengujian model pertama tidak memberikan tingkat keberhasilan seperti yang diinginkan dalam Tujuan Perancangan ini, maka dapat dipastikan simulator tidak layak untuk diuji lagi. Sebaliknya bila hasil pengujian model pertama memberikan hasil yang signifikan, maka pengujian terhadap model kedua perlu dilakukan untuk mengetahui perilaku simulator terhadap kondisi data lainnya.

VARIABEL

Variabel-variabel yang digunakan dalam penelitian ini secara garis besar dibagi menjadi dua bagian, yaitu variabel *input* dan variabel *output*. Secara konseptual, semua variabel adalah merupakan indikator pada analisa teknikal. Hanya dalam penelitian ini, variabel tersebut dimodifikasi tanpa menghilangkan fungsinya sehingga dapat dimasukkan ke dalam sistem yang dirancang. Variabel-variabel dipilih berdasarkan pengamatan grafik dan juga penelitian-penelitian yang telah terlebih dahulu dilakukan baik yang menggunakan metode *fuzzy* atau pun tidak. Berikut adalah penjelasan untuk masing-masing variabel, yaitu:

- Variabel persentase perubahan volume. Variabel ini digunakan dalam rancangan simulator untuk menunjukkan tingkat signifikansi perubahan variabel lainnya. Asumsi yang digunakan adalah bahwa perubahan volume yang besar menunjukkan adanya perubahan pola perilaku terhadap variabel lainnya.
- Variabel persentase perubahan harga terdiri atas perubahan pada hari perhitungan (d), perubahan pada sehari sebelumnya ($d-1$), dan perubahan dua pada hari sebelumnya ($d-2$). Ketiga variabel ini telah digunakan dalam penelitian oleh J. Fat serta C. D. Souto-Maior, F. D. Murcia, J. A. Borba dan N. D. Costa Jr.
- Variabel persentase perubahan MA15, MA50 dan MA200. Variabel ini digunakan dalam bentuk sederhananya yang dikenal dengan istilah *Simple Moving Average* (SMA). Penggunaan variabel MA15, MA50 DAN MA200 dilakukan penyesuaian. Ketiga variabel ini telah banyak dimanfaatkan oleh para peneliti dengan berbagai modifikasi untuk penyesuaian terhadap sistem, seperti pada S. Agrawal, M. Jindal dan G. N. Pillai; A. A. Gamil, R. S. El-fouly dan N. M. Darwish; dan V. Olej. Juga variabel ini ditemukan dalam literatur seperti yang ditulis oleh Z. Bodie, A. Kane dan A. J. Marcus.

VARIABEL PERSENTASE PERUBAHAN SMA

Periode SMA yang digunakan adalah 15, 50 dan 200. Nilai SMA ini dihitung berdasarkan Persamaan (1).

$$SMA_n = \frac{x_1 + x_2 + x_3 + \dots + x_n}{n} \dots\dots\dots(1)$$

dengan nilai n menentukan periode SMA yang akan dihitung, nilai $x_1, x_2, x_3, \dots, x_n$ merupakan data harga penutupan pada hari ke-1, 2, 3 hingga hari ke- n . Nilai-nilai SMA yang telah dihitung berdasarkan persamaan (2), akan dijadikan dasar untuk menghitung variabel persentase perubahan SMA periode n ($\%dSMA_n$). Dengan nilai n merupakan nilai periode 15, 50 dan 200. Variabel $\%dSMA_n$ inilah yang akan dijadikan nilai *input* bagi sistem. Persamaan (2) digunakan untuk menghitung nilai variabel ini.

$$\%dSMA_n = \frac{SMA_n - SMA_n^{-1}}{SMA_n} \times 100\% \dots\dots\dots(2)$$

dengan nilai SMA_n merupakan nilai SMA periode ke- n pada satu hari sebelum hari perhitungan, nilai SMA_n^{-1} merupakan nilai SMA periode ke- n pada dua hari sebelum hari perhitungan.

VARIABEL PERSENTASE PERUBAHAN VOLUME

Nilai variabel persentase perubahan volume dihitung berdasarkan persamaan (3). Persamaan ini menghitung nilai perubahan berdasarkan selisih antara volume pada satu hari sebelum hari perhitungan dengan

nilai volume pada dua hari sebelum hari perhitungan. Apabila nilai volume satu hari sebelum hari perhitungan adalah nol (dalam arti hari tersebut merupakan hari libur bursa sehingga tidak ada transaksi), maka nilai yang digunakan adalah dua hari sebelum hari perhitungan dikurangkan dengan nilai volume tiga hari sebelum hari perhitungan. Ini dilakukan untuk menghindari timbulnya pembagian terhadap nilai nol.

$$\%dVol = \frac{Vol_{n-1} - Vol_{n-2}}{Vol_{n-1}} \times 100\% \dots\dots\dots(3)$$

VARIABEL PERSENTASE PERUBAHAN HARGA

Harga yang digunakan pada perhitungan ini adalah harga penutupan pada sesi II transaksi di Bursa Efek Indonesia. Nilai perubahan yang dihitung adalah perubahan harga satu hari sebelum hari perhitungan terhadap dua hari sebelumnya (dP_{-1}), perubahan harga dua hari sebelum hari perhitungan terhadap tiga hari sebelumnya (dP_{-2}) dan perubahan harga pada hari perhitungan terhadap satu hari sebelumnya (dP_0). Nilai-nilai variabel ini dihitung dengan persamaan (4).

Nilai yang menjadi *input* dan *output* bagi sistem ini adalah nilai persentase perubahan variabel-variabel tersebut (masing-masing variabel adalah $\%dP_{-1}$, $\%dP_{-2}$, dan $\%dP$), yang dihitung mengikuti persamaan (5). Variabel *output* $\%dP$ dihitung semata-mata adalah untuk keperluan perancangan dan pelatihan sistem. Untuk sistem berjalan, nilai $\%dP$ ini adalah target perhitungan sistem.

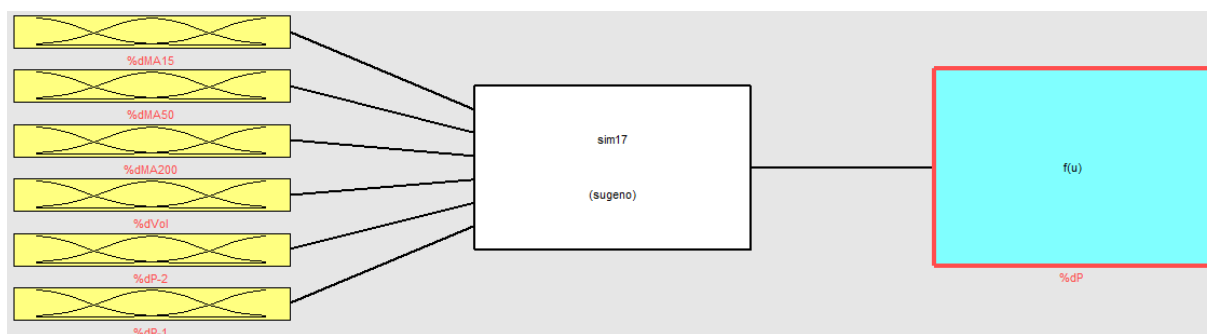
$$dP_n = P_n - P_{n-1} \dots\dots\dots(4)$$

$$\%dP_n = \frac{dP_n}{P_n} \times 100\% \dots\dots\dots(5)$$

dengan n berupa nilai 0 (hari perhitungan), -1 (satu hari sebelum hari perhitungan), dan -2 (dua hari sebelum hari perhitungan).

RANCANGAN FIS

Simulator ANFIS ini dirancang berdasarkan diagram pada Gambar 3 berikut ini. Blok kuning menunjukkan proses fuzzifikasi variabel-variabel *input*. Blok putih menunjukkan proses inferensi yang menggunakan metode Sugeno berdasarkan aturan-aturan yang telah ditetapkan. Blok hijau menunjukkan hasil *output* dengan metode defuzzifikasi *weighted average*.



■ Gambar 3. Diagram Blok Simulator ANFIS

Simulator FIS ini di-*generate* dengan menggunakan metode *Grid Partition*. FIS yang di-*generate* ini bertipe Sugeno. Variabel-variabel difuzzifikasi dengan menggunakan fungsi keanggotaan Gauss. Masing-masing variabel memuat tiga macam fungsi keanggotaan tipe Gauss ini. Dari hasil percobaan yang telah dilakukan, diketahui bahwa dengan menggunakan tiga macam fungsi keanggotaan tipe Gauss untuk tiap variabel akan memberikan hasil prediksi yang efektif. Fungsi keanggotaan tipe Gauss ini akan dihitung berdasarkan Persamaan (6).

$$f(x; \sigma, c) = e^{-\frac{(x-c)^2}{2\sigma^2}} \dots\dots\dots(6)$$

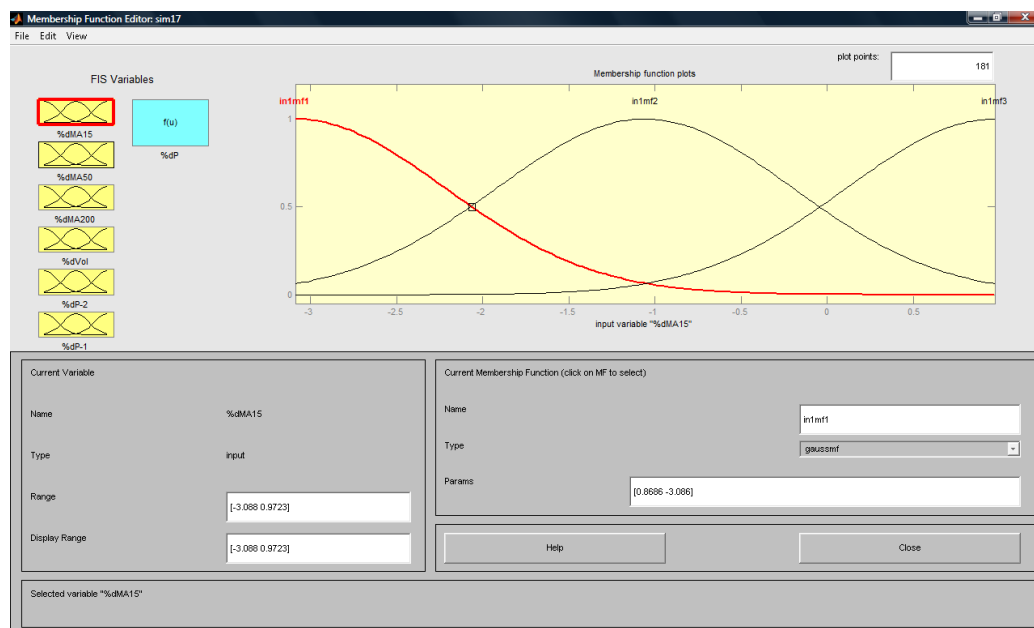
Variabel *output* $\%dP$ menggunakan fungsi keanggotaan bertipe konstanta. *Rule* dalam simulator ini juga ditentukan sesuai dengan hasil pelatihan oleh NN. Hasil rancangan simulator sesuai dengan masing-masing momen memiliki karakteristik sebagai berikut:

- Enam variabel *input*, yaitu %dMA15, %dMA50, %dMA200, %dVol, %dP₋₂, dan %dP₋₁. Masing-masing variabel *input* memiliki tiga label fungsi keanggotaan bertipe Gauss (Persamaan (6)).
- Satu variabel *output*, yaitu %dP. Variabel *output* ini bertipe konstanta dengan label sebanyak 729 buah.
- Jumlah *rules* adalah 729 buah, yang masing-masing bertipe pernyataan berkondisi IF-THEN-ELSE.
- Operator logika ekuivalen yang digunakan adalah sebagai berikut:
 - Operator logika AND menggunakan metode *product*.
 - Operator logika OR menggunakan metode *probabilistic OR* (*probor*).
 - Operator logika Implikasi menggunakan aturan minimum.
- Aturan agregasi yang digunakan adalah maksimum. Aturan agregasi ini akan menggabungkan seluruh aturan yang telah dieksekusi.
- Metode defuzzifikasi yang digunakan *weighted average* (*wtaver*).

Keempat simulator hasil rancangan memiliki karakteristik tersebut, hanya berbeda dalam rentang nilai pada masing-masing fungsi keanggotaan atau pun konstanta *output*. Diagram rancangan adalah seperti yang telah ditunjukkan pada Gambar 3.

RANCANGAN VARIABEL INPUT

Rancangan variabel *input* untuk simulator *Bearish* dapat dilihat pada Gambar 4. Seperti yang telah dijelaskan, keempat simulator yang dirancang ini memiliki karakteristik *input* yang sama, hanya berbeda pada nilai *range*.



■ Gambar 4. Simulator *Bearish*: Variabel Input %dMA15

RANCANGAN VARIABEL OUTPUT

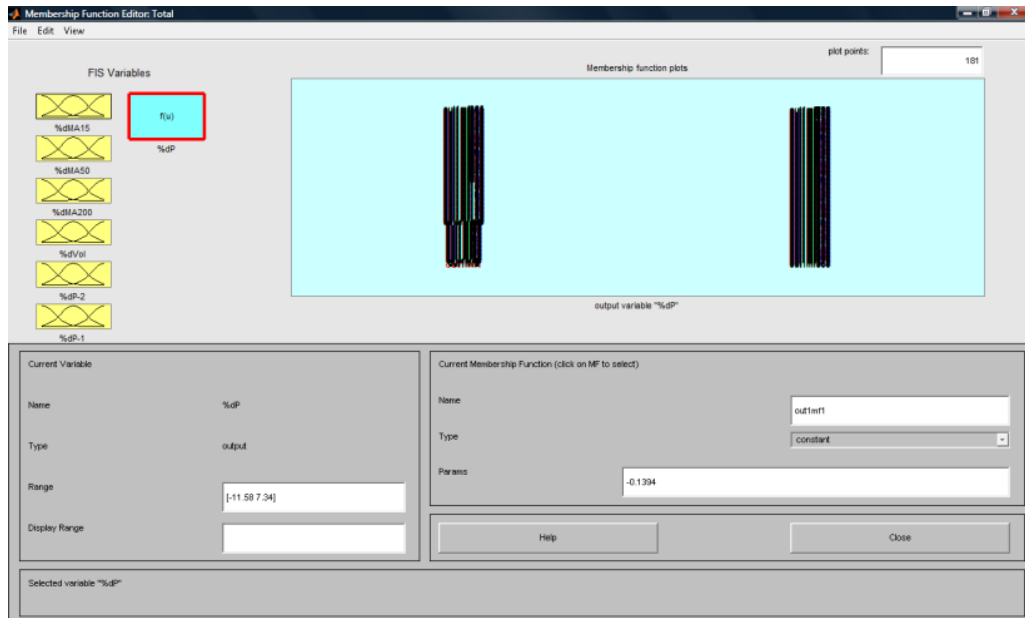
Rancangan variabel *output* keempat sistem simulator ini adalah variabel %dP. Variabel ini menunjukkan nilai perubahan yang diperkirakan oleh sistem pada hari perhitungan. Variabel ini dirancang bertipe konstanta dengan 729 label. Rancangan variabel *output* %dP untuk simulator *total* ditunjukkan oleh Gambar 5. *Output* ini bukanlah kesimpulan dari sistem. Nilai-nilai yang dihasilkan oleh *output* ini masih memerlukan proses lebih lanjut dan pada akhirnya memerlukan interpretasi dari perancang [10].

RANCANGAN RULES

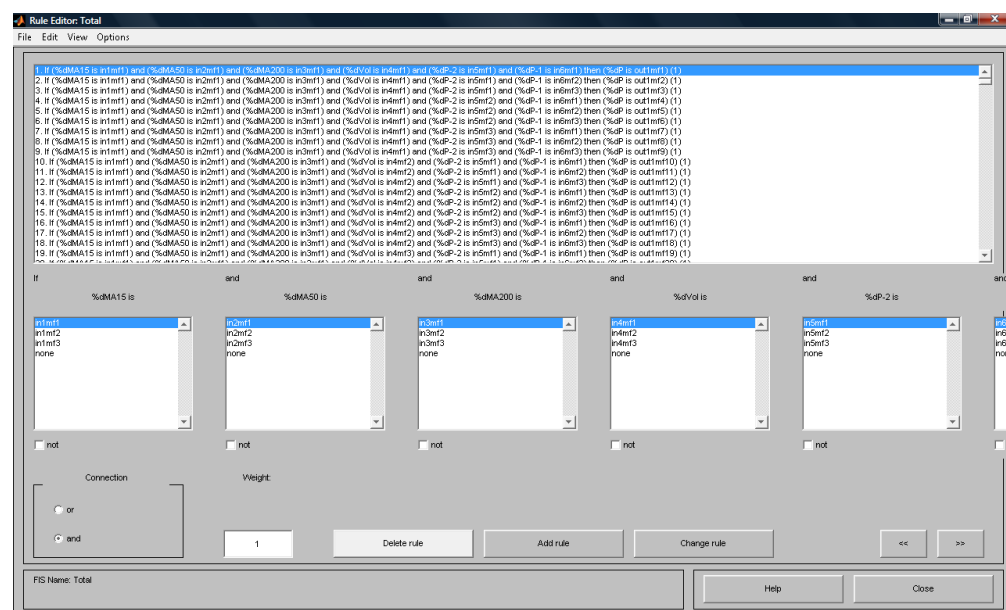
Rules yang dirancang untuk masing-masing simulator adalah sebanyak 729 aturan. Rancangan aturan-aturan ini menggunakan pernyataan berkondisi IF-THEN-ELSE. Hubungan antar-variabel *input* dalam aturan tersebut menggunakan operator OR atau AND. Bobot yang digunakan untuk tiap aturan adalah 1. Gambar 6 memperlihatkan rancangan peraturan untuk simulator *total*.

RANCANGAN NN: MOMEN BEARISH

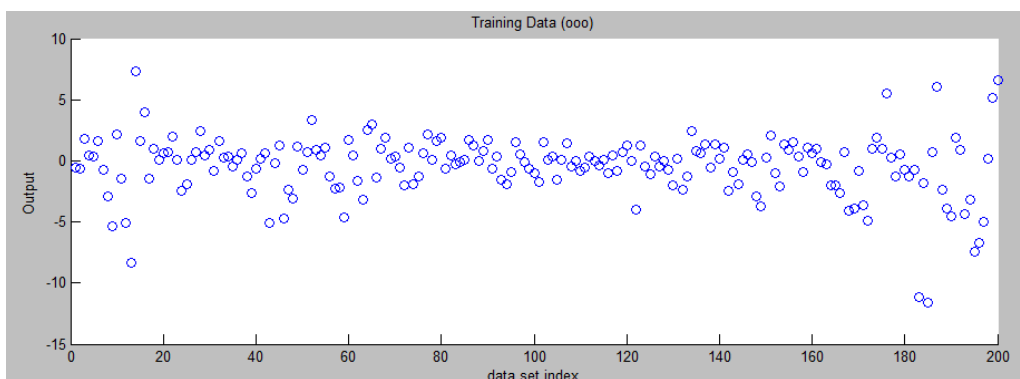
Data yang akan dilatih berjumlah 200 entri. NN akan mengestimasi FIS sehingga diharapkan memiliki kemampuan yang lebih baik dalam memprediksi hasil yang *output* yang diinginkan. Gambar 7 memperlihatkan entri data yang telah di-plot pada grafik *training data*. Setiap entri data ini terdiri dari enam variabel *input* dan satu variabel *output*.



■ **Gambar 5.** Rancangan Variabel Output %dP untuk Simulator Total



■ **Gambar 6.** Rancangan Rules untuk Simulator Total



■ **Gambar 7.** Data Pelatihan NN untuk Momen Bearish

Metode untuk menghasilkan FIS adalah metode *Grid Partition*, dengan ketentuan sebagai berikut:

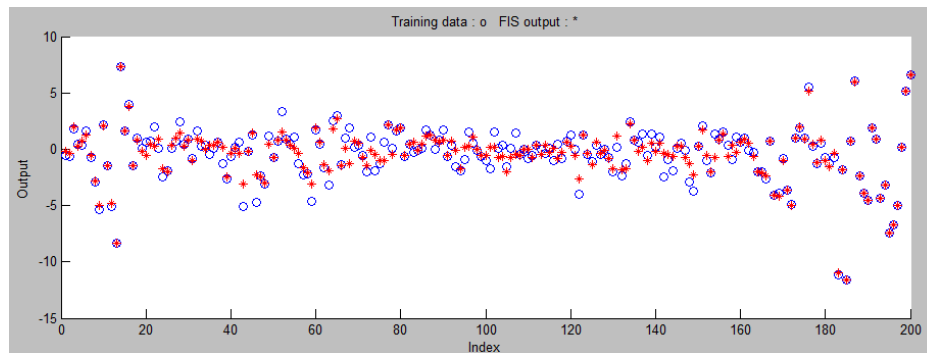
- Jumlah variabel *input* adalah enam buah, dengan masing-masing tiga buah label fungsi keanggotaan,
- Fungsi keanggotaan berjenis *Gauss*,
- Jumlah variabel *output* adalah satu buah dengan tipe fungsi keanggotaan berupa konstanta.

FIS yang telah dihasilkan ini, akan dilatih dengan menggunakan metode *hybrid*, dengan toleransi kesalahan nol dan *epoch* maksimum 30 langkah. Hasil pelatihan menunjukkan kestabilan penurunan nilai

kesalahan pada *epoch* ke-25. Hasil pelatihan sistem ANFIS yang dirancang ini dapat dilihat pada Gambar 8. Tanda bulatan biru menunjukkan entri data yang dilatih, sedangkan tanda bintang merah menunjukkan *output* hasil perhitungan sistem.

RANCANGAN NN: MOMEN *SIDEWAYS*

Sistem ANFIS momen *Sideways* ini akan dilatih dengan data yang berjumlah 99 entri. Metode untuk menghasilkan FIS momen *Sideways* serupa dengan metode momen *Bearish*. Karakteristik yang digunakan juga tepat sama. FIS yang telah dihasilkan ini, akan dilatih dengan menggunakan metode *hybrid*, dengan toleransi kesalahan nol dan *epoch* maksimum 5 langkah. Nilai kesalahan sistem hasil akhir pelatihan adalah 0.0075.



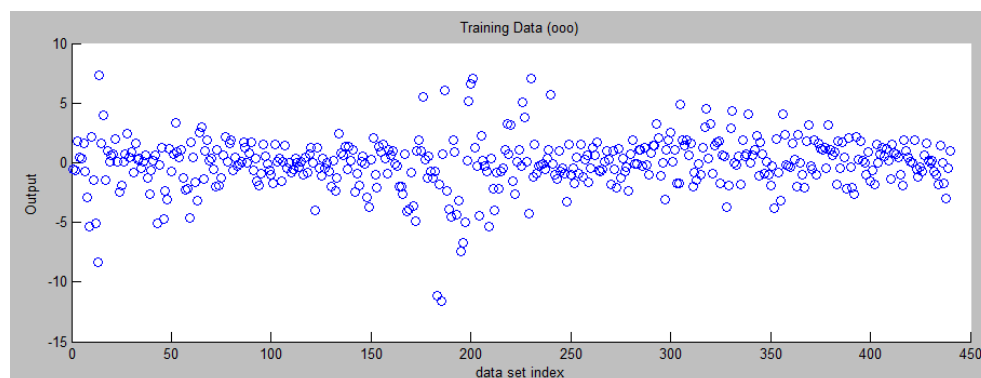
■ Gambar 8. Hasil Pelatihan Sistem ANFIS Momen *Bearish*

RANCANGAN NN: MOMEN *BULLISH*

Jumlah data pelatihan untuk momen *Bullish* ini adalah sebanyak 143 data. Metode untuk menghasilkan ANFIS adalah dengan menggunakan metode *Grid Partition*. Karakteristik sistem adalah sama dengan momen lainnya. FIS yang telah dihasilkan ini, akan dilatih dengan menggunakan metode *hybrid*, dengan toleransi kesalahan nol dan *epoch* maksimum 15 langkah. Hasil pelatihan menunjukkan kestabilan penurunan nilai kesalahan pada *epoch* ke-11. Nilai kesalahan sistem hasil akhir pelatihan adalah 0.021.

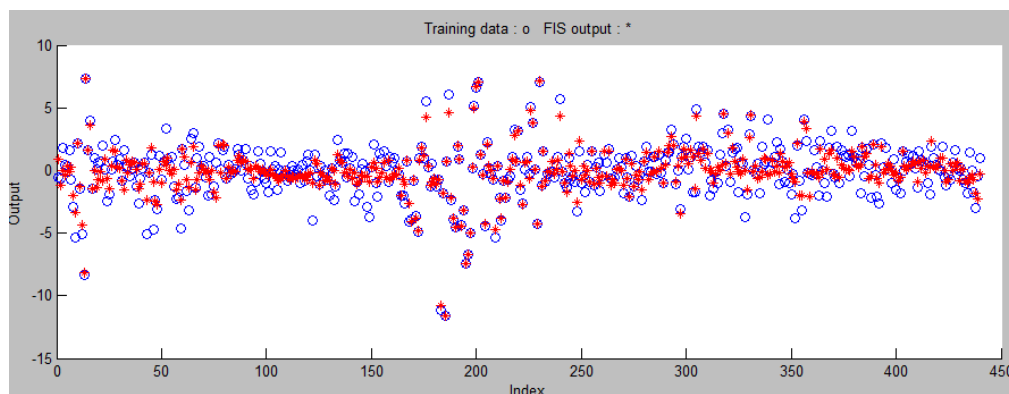
RANCANGAN NN: TOTAL

Simulator untuk *Total* ini adalah simulator yang diharapkan dapat bekerja terhadap semua momen. Jumlah data pelatihan untuk momen simulator ini adalah sebanyak 442 data. Ini adalah keseluruhan data yang digunakan dalam perancangan ini. Gambar 9 memperlihatkan seluruh entri data tersebut. Metode yang digunakan untuk menghasilkan ANFIS masih menggunakan metode *Grid Partition* dengan karakteristik yang sama dengan simulator lainnya.



■ Gambar 9. Data Pelatihan NN untuk *Total*

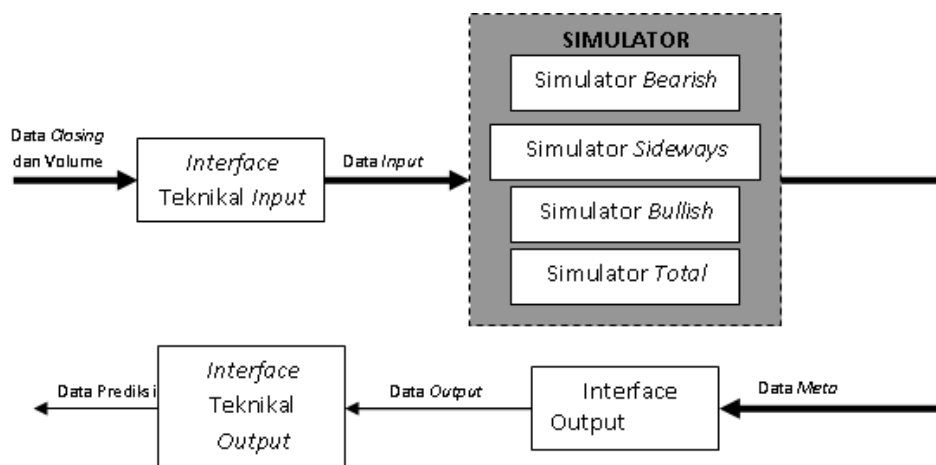
FIS yang telah dihasilkan ini, akan dilatih dengan menggunakan metode *hybrid*, dengan toleransi kesalahan nol dan *epoch* maksimum 25 langkah. Hasil pelatihan menunjukkan kestabilan penurunan nilai kesalahan pada *epoch* ke-9. Nilai kesalahan sistem hasil akhir pelatihan adalah 1.012. Struktur ANFIS hasil rancangan menggunakan metode yang sama dengan struktur momen lain. Data yang digunakan untuk pengecekan awal simulator adalah sama dengan data yang digunakan untuk pelatihan. Ini bertujuan sebagai pembuktian awal atas efektivitas simulator. Gambar 10 menunjukkan hasil pengecekan awal simulator *Total* terhadap data pelatihan.



■ Gambar 10. Hasil Pelatihan Sistem ANFIS Total

KONSEP PENGUJIAN SIMULATOR

Diagram pada Gambar 11 memperlihatkan konsep pengujian terhadap simulator-simulator yang telah dirancang. Data JKSE histori yang diperlukan adalah minimum sebanyak 200 entri sebelum hari perhitungan. Data tersebut adalah nilai penutupan indeks dan besar volume.



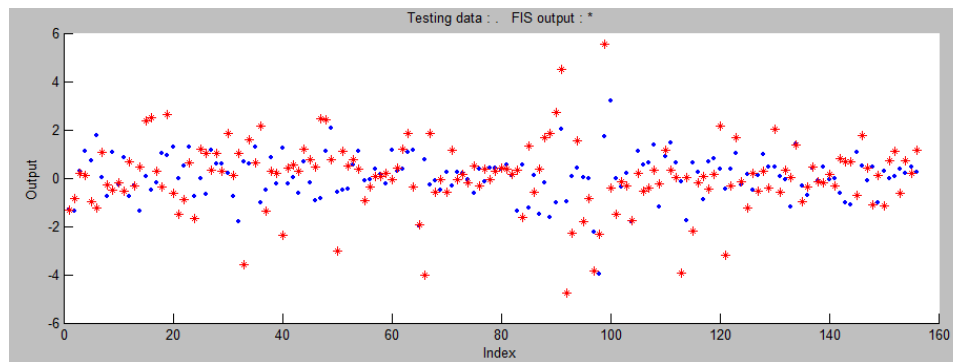
■ Gambar 11. Konsep Pengujian Simulator

Selanjutnya, data tersebut diumpankan ke modul *Interface Teknikal Input*. Oleh modul *interface*, data nilai penutupan dan volume akan dihitung dan dikonversi menjadi data nilai variabel *input* bagi simulator. Data inilah yang merupakan data *input* bagi simulator. Modul *Interface Output* membantu pemanggilan simulator. Tanpa menggunakan modul ini, pemanggilan simulator dalam pengujian akan menjadi tidak efisien. Hasil dari *Interface Output* merupakan nilai perhitungan oleh simulator, yang dalam rancangan ini dinamakan sebagai data *output*. Data *output* tersebut diumpankan ke modul *Interface Teknikal Output*. Hasil dari modul *interface* ini merupakan data prediksi yang dapat dipahami oleh *user*. *Interface Output* yang digunakan dalam pengujian ini adalah berupa *script* yang dijalankan secara otomatis. Pada prinsipnya, tujuan dari pengujian ini adalah untuk melihat apakah simulator-simulator memiliki tingkat ketepatan seperti yang diinginkan.

PENGUJIAN DAN ANALISIS SIMULATOR: MOMEN BEARISH

Hasil secara visual ini juga didukung hasil perhitungan secara statistik. Dari 200 entri data, simulator momen *Bearish* mampu memprediksi dengan tepat sebanyak 173 kali. Ini berarti tingkat ketepatan simulator menghitung kembali data pelatihan adalah 86,5%. Angka 86,5% ini menunjukkan tingkat ketepatan simulator dalam memprediksi arah pergerakan indeks. Sebaliknya, simulator mampu menghitung nilai perubahan indeks dengan ketepatan 98,7%. Dengan hasil yang demikian ini, dapat disimpulkan simulator mampu menghitung dan memprediksi kembali data pelatihannya dengan memuaskan.

Pengujian terhadap data pengecekan dilakukan secara seragam terhadap semua simulator. Grafik hasil pengujian terhadap data pengecekan oleh simulator *Bearish* dapat dilihat pada Gambar 12. Tanda titik biru menunjukkan entri data yang ingin diuji, sedangkan tanda titik merah merupakan hasil pengujian oleh simulator. Secara garis besar, grafik memperlihatkan sistem mampu memprediksi arah gerak indeks, tetapi dengan persentase ketepatan yang rendah. Hasil perhitungan secara statistik juga memperlihatkan bahwa tingkat ketepatan sistem memprediksi arah hanya 50,64%. Nilai ini jauh lebih rendah dari pada tingkat ketepatan yang diinginkan dalam Tujuan Perancangan. Tingkat kesalahan prediksi nilai *output* secara rata-rata adalah 8,94%.

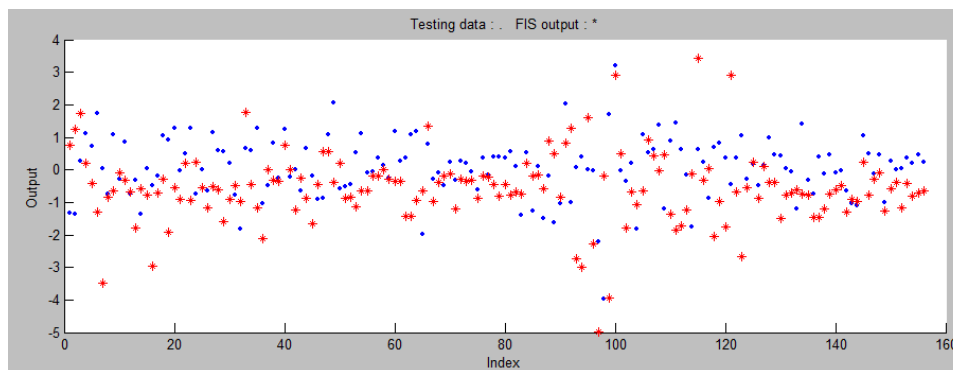


■ Gambar 12. Grafik Pengujian Data Pengecekan Simulator *Bearish*

PENGUJIAN DAN ANALISIS SIMULATOR: MOMEN *SIDWAYS*

Gambaran visual memperlihatkan bahwa simulator memang memiliki kemampuan untuk memprediksi dan menghitung *output* sistem dengan tingkat ketepatan melebihi tujuan rancangan sistem. Melalui perhitungan secara statistik, didapatkan fakta yang memperkuat dugaan ini. Simulator mampu memprediksi arah pergerakan *output* dengan nilai ketepatan 99,3%. Hasil perhitungan terhadap besarnya perubahan yang terjadi adalah 99,8%. Ini berarti simulator mampu menghitung besar perubahan hanya dengan faktor kesalahan sekitar 0.13%. Hasil ini juga memperlihatkan bahwa simulator momen *sideways* ini berhasil dalam pengujian model pertama.

Pengujian data pengecekan menjadi suatu keharusan terhadap simulator *Sideways*. Gambar 13 memperlihatkan hasil *plot* hasil pengujian simulator *Sideways* dengan menggunakan data pengecekan.



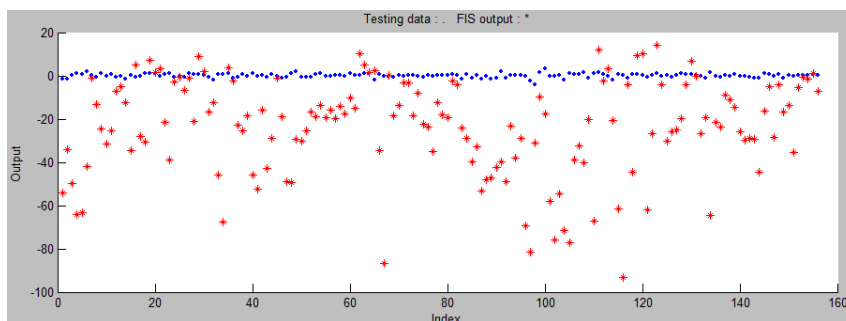
■ Gambar 13. Grafik Pengujian Data Pengecekan Simulator *Sideways*

Banyaknya penyimpangan titik merah terhadap titik biru pada Gambar 13 memperlihatkan bahwa sangat besar simulator *Sideways* gagal memprediksi arah mau pun nilai indeks pada data pengecekan. Ini terlihat dari hasil perhitungan berikut. Tingkat ketepatan prediksi arah adalah 50,64%, dan tingkat kesalahan perhitungan nilai *output* adalah 182,51%. Ini menunjukkan simulator *Sideways* tidak dapat digunakan untuk memprediksi entri data dari momen yang acak.

PENGUJIAN DAN ANALISIS SIMULATOR: MOMEN *BULLISH*

Gambar visual menunjukkan sekilas bahwa simulator mampu memprediksi data *output* pelatihannya dengan baik. Ini ditandai dengan banyaknya bulatan biru yang berhimpitan dengan tanda bintang merah. Secara statistik, hasil simulasi mampu memprediksi arah hingga tingkat ketepatan 98,9%. Rata-rata kesalahan perhitungan terhadap nilai *output* sebenarnya adalah 0,0252%. Dengan kata lain, simulator *Bullish* juga mampu melewati pengujian model pertama. Baik tingkat ketepatan prediksi arah mau pun ketepatan perhitungan *output* melebihi ekspektasi.

Dari analisis terhadap dua simulator di atas terhadap pengujian data pengecekan, hampir dapat dipastikan simulator *Bullish* akan gagal. Ini terkait dengan pelatihan yang diterima oleh masing-masing simulator. Data yang digunakan untuk pelatihan adalah data yang homogen dengan suatu kecenderungan tertentu. Oleh karenanya, ketika diumpankan data yang sifatnya acak (tanpa diketahui jenis momennya terlebih dahulu), simulator gagal baik dalam ketepatan menghitung arah mau pun nilainya. Gambar 14 memperjelas hasil pengujian simulator *Bullish* terhadap data pengecekan. Gambar ini dengan jelas memperlihatkan kegagalan simulator. Hasil perhitungan menunjukkan tingkat ketepatan arah simulasi hanyalah 46,79%. Dengan persentase kesalahan perhitungan nilai *output* adalah 8,33%.



■ **Gambar 14.** Grafik Pengujian Data Pengecekan Simulator *Bullish*

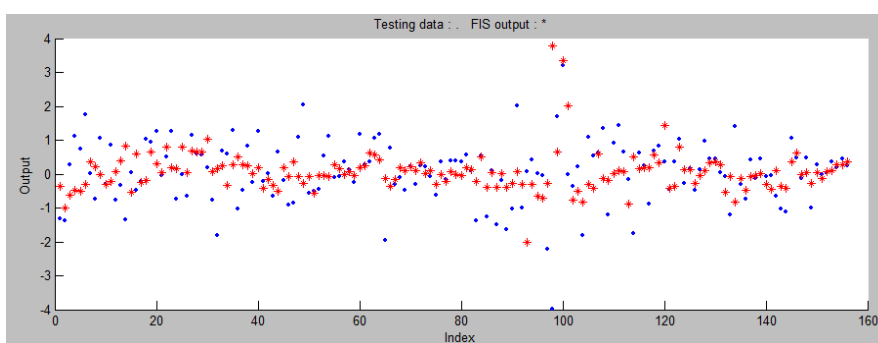
PENGUJIAN DAN ANALISIS SIMULATOR: TOTAL

Simulator ini merupakan simulator yang diharapkan dapat mengakomodasi seluruh momen yang ada, atau dengan kata lain simulator mampu melakukan prediksi tanpa perlu mengetahui momen yang sedang berlangsung. Hasil pengujian model pertama terhadap simulator ini sedikit menyulitkan untuk ditarik kesimpulan secara visual

karena banyaknya data yang diuji. Oleh sebab itu, mutlak diperlukan perhitungan secara statistik. Tingkat ketepatan prediksi arah dengan menggunakan simulator ini adalah 76,13%.

Angka ini sangat jauh di bawah simulator lainnya. Ini dapat dipahami karena simulator lain menggunakan data yang seragam. Artinya, data tersebut memiliki pola yang relatif sama, seperti contoh pada momen *bearish*. Data pada momen *bearish* dipastikan memiliki pola penurunan yang jauh lebih signifikan dibandingkan kenaikan, sebaliknya juga momen *bullish*. Dengan demikian, hasil pengujian dengan data pelatihan yang sama, simulator-simulator selain *Total* ini secara wajar harus menunjukkan hasil yang sangat tinggi. Kemampuan simulator *Total* dalam menghitung nilai *output* juga terlihat jauh di bawah kemampuan simulator lain, yaitu pada tingkat kesalahan perhitungan 4,51%. Hasil ini juga dapat dipahami sesuai dengan prinsip keseragaman data yang telah dijelaskan di atas.

Dengan kegagalan seluruh simulator momen yang ada, simulator *Total* memperlihatkan unjuk kerja seperti yang diharapkan. Gambar 15 memperlihatkan hasil *plotting* pengujian terhadap data pengecekan. Posisi titik merah terhadap data pengecekan titik biru, menunjukkan potensi yang cukup bagus, walau pun masih terlalu dini untuk mengatakan bahwa simulator memang mampu memprediksi. Oleh karena itu, perhitungan dengan menggunakan *Interface Teknikal Output* mutlak diperlukan. Hasil perhitungan ini menunjukkan sistem mampu memprediksi arah dengan tingkat ketepatan sebesar 61,55%, dan tingkat kesalahan perhitungan nilai *output* adalah sebesar 4,5%.



■ **Gambar 15.** Grafik Pengujian Data Pengecekan Simulator *Total*

KESIMPULAN

Hasil pengujian memperlihatkan bahwa keempat simulator yang dirancang (simulator *Bearish*, *Sideways*, *Bullish* dan *Total*) mampu memberikan prediksi dan perhitungan nilai *output* yang jauh di atas Tujuan Rancangan terhadap data pelatihannya sendiri. Tetapi dari keempat simulator ini, hanya simulator *Total* yang memperlihatkan unjuk kerja sesuai Tujuan Rancangan terhadap data pengecekan. Oleh karena itu, dapat disimpulkan bahwa penelitian ini berhasil merancang simulator sesuai dengan Tujuan Rancangan. Dalam hal ini adalah simulator *Total* (tingkat kebenaran prediksi arah 61,55% dengan 156 entri data, mencakup ketiga momen secara keseluruhan).

Simulator *Bearish* (50,64%), *Sideways* (50,64%) dan *Bullish* (46,79%) gagal dalam pengujian terhadap data pengecekan karena ketiga simulator ini dilatih dengan menggunakan data yang homogen. Kehomogenan data pelatihan masing-masing simulator ini terkait dengan momen dari mana data tersebut dicuplik. Ketiga

simulator ini pada prinsipnya bukanlah kegagalan sistem, tetapi juga berfungsi sebagai metode untuk mengetahui secara keseluruhan kemampuan belajar NN dalam membangun sistem FIS. Hasil penelitian ini justru memperlihatkan bahwa NN dapat digunakan untuk memperbaharui batas-batas nilai keanggotaan masing-masing label pada variabel FIS.

DAFTAR PUSTAKA

- [1] M. Alizadeh, M. Gharakhani, E. Fotoohi and R. Rada, *Design and Analysis of Experiments in ANFIS Modeling for Stock Price Prediction*, International Journal of Industrial Engineering Computations, 2011, p. 409.
- [2] J. Fat, *Prediksi Arah Indeks Harga Saham Gabungan dengan Menggunakan Metode Fuzzy Logic*, Seminar Nasional Ritektra, 2010, hal. 93.
- [3] J. Fat, *Perancangan Simulasi Sistem Prediksi Arah Gerak Harga Saham Berbasis Analisis Teknikal dengan Fuzzy Inference System*, Universitas Tarumanagara, Jakarta, 2011, hal. 37.
- [4] W. K. Wong, M. Manzur and B. K. Chew, *How Rewarding is Technical Analysis? Evidence from Singapore Stock Market*, Working Paper No. 0216, Department of Economics, National University of Singapore, 2002.
- [5] L. A. Zadeh, *Fuzzy Logic=Computing with Words*, IEEE Transactions on Fuzzy Systems, Vol. 4, No. 2, May 1996, p. 103.
- [6] L. A. Zadeh, *Toward Human Level Machine Intelligence-Is It Achievable? The Need for a Paradigm Shift*, International Journal of Advanced Intelligence, Vol. 1, No. 1, October 2009, p. 3.
- [7] S. M. Sivanandam, S. Sumathi and S. N. Deepa, *Introduction to Fuzzy Logic using Matlab*, New York: Springer-Verlag Berlin Heidelberg, 2007, pp. 81-84.
- [8] J. F. Weston and E. F. Brigham, *Essentials Of Managerial Finance*, Fort Worth: The Dryden Press, 1993, pp. 26-27.
- [9] M. Sewell, *Behavioral Finance*. University Of Cambridge, Februari 2007 (revisi April 2010), pp. 1, 6-7.
- [10] L. A. Zadeh, *Outline of A New Approach to the Analysis of Complex Systems and Decision Processes*, IEEE Transactions on Systems, Man, and Cybernetics Vol. SMC-3 No. 1, January 1973, p. 39.