

OPTIMASI PERTUMBUHAN TANAMAN KETIMUN MINI (*Cucumis sativus* L. Var. Marla) PADA FASE VEGETATIF DENGAN ALGORITMA GENETIKA

*Optimization of Vegetative Stage of Baby Cucumber
(Cucumis sativus L. Var. Marla) Using The Genetic Algorithm*

Tamrin¹, K. B. Seminar², H. Suhardiyanto³, S. Hardjoamidjodjo⁴

Abstract

*An optimization was conducted to a growth model for baby cucumber (*Cucumis sativus* L. Var. Marla). The model gives information of nutrient intake and micro climate conditions (temperature, humidity, and irradiation) to get the response of the plant (ratio of canopy area and stem diameter) and the loss of nutrient solution. The model was then coupled with genetic algorithm in order to get optimum result efficiently. Two fitness functions were used in the application of genetic algorithm. The results showed that the first fitness function could be used to optimize the plant response (ratio of canopy area and stem diameter) and the loss of nutrient solution efficiently as indicated by the maximum fitness value ($P_c = 0.6$ and $P_m = 0.01$).*

Keywords: *genetic algorithm, artificial neural networks, plant response, nutrient solution, micro climate*

PENDAHULUAN

Secara fisik, faktor lingkungan yang mempengaruhi tanaman dalam rumah tanaman dapat diklasifikasikan menjadi dua bagian (Esmay dan Dixon, 1986), yaitu faktor lingkungan udara sekitar, di bagian atas tanaman dan faktor lingkungan pada media tumbuh tanaman, di bagian bawah tanaman. Faktor lingkungan udara sekitar meliputi suhu, cahaya, kelembaban, dan CO₂, sedangkan faktor lingkungan di media tumbuh, meliputi suhu tanah, kadar air tanah, pH, nutrisi, konduktivitas listrik, evaporasi dan lain-lain (tergantung sistem budidaya).

Apabila dapat menciptakan keadaan lingkungan yang lebih sesuai dengan kebutuhan tanaman maka dapat diperoleh beberapa keuntungan, antara lain adalah produksi yang sesuai dengan yang diinginkan, penghematan energi dan ramah lingkungan. Selain itu, juga memungkinkan untuk dikembangkan lebih lanjut dengan cara memanipulasi lingkungan sehingga didapat produk berkualitas dan dengan nilai tambah yang lebih tinggi.

Salah satu metode meminimalkan pengaruh lingkungan terhadap tanaman adalah menggunakan teknologi rumah tanaman. Didalam rumah tanaman, faktor

Staf Pengajar Program Studi Teknik Pertanian, FAPERTA-UNSRI

234 Staf Pengajar Departemen Teknik Pertanian, FATETA-IPB

lingkungan lebih mungkin dikendalikan untuk mendapatkan kondisi pertumbuhan yang lebih baik. Salah satu alasan penggunaan rumah tanaman adalah menyediakan sedikit modifikasi terhadap iklim alam yang ada.

Karakteristik gen tertentu suatu tanaman atau kondisi produk yang diinginkan tidak akan muncul seperti yang diharapkan bila tidak didukung oleh penyediaan kondisi lingkungan yang sesuai. Namun demikian, sebelum dilakukan upaya pengontrolan terhadap kondisi lingkungan tersebut, perlu ditentukan terlebih dahulu acuan optimum (optimal setpoint) dari lingkungan tersebut, sehingga upaya pengontrolan mencapai tujuan yang diinginkan. Salah satu teknik optimasi yang makin banyak digunakan akhir-akhir ini adalah algoritma genetika. Penggunaannya dalam kontrol greenhouse, diantaranya dikemukakan oleh Ursem *et al.* (2002). Algoritma genetika menggunakan analog fenomena natural, yaitu adaptasi evolusi biologis, dimana individu-individu terbaik dalam suatu populasi akan mengalami persilangan-persilangan dan mutasi-mutasi dimana yang lebih baik dapat bertahan, sedangkan yang lemah akan punah.

Populasi terdiri dari individu-individu yang masing-masing mempresentasikan penyelesaian yang mungkin untuk suatu permasalahan. Masing-masing individu, yang dalam hal ini dipersamakan dengan kromosom, mempunyai nilai fitness yang bersesuaian dengan kelayakan solusi permasalahan tersebut.

Beberapa individu dalam populasi dengan nilai fitness lebih baik berpeluang untuk beriterasi (reproduksi). Dalam prosesnya akan terjadi rekombinasi (*cross over*) dan mutasi. Setelah mengalami seleksi akan menghasilkan individu-individu baru yang diharapkan merupakan sekumpulan solusi yang paling mungkin, yang pada gilirannya iterasi dihentikan dan dipilih satu solusi yang paling mungkin.

Salah satu kelebihan algoritma genetika adalah relatif sederhana karena kemampuannya untuk belajar dan beradaptasi, yaitu hanya memerlukan informasi tentang struktur kromosom (individu) dan bentuk fungsi *fitness* dari permasalahan yang dihadapi kemudian akan mencari sendiri solusi terbaik (Yandra dan Hermawan, 2000).

Model jaringan syaraf tiruan (JST) untuk pertumbuhan tanaman ketimun mini, dengan cara simulasi, telah dapat ditentukan nilai optimum dari rasio kanopi-diameter batang tanaman dan limpasan. Namun demikian penentuan secara simulasi kurang efisien (Tamrin *et al.*, 2005)

Sehubungan dengan hal di atas, pada makalah ini akan dipaparkan model jaringan syaraf tiruan (JST) yang menggambarkan hubungan volume larutan nutrisi yang diberikan, suhu, kelembaban dan iradiasi dengan rasio luas kanopi-diameter batang (tanaman) dan larutan nutrisi yang hilang (limpasan), yang akan dilengkapi dengan teknik optimasi algoritma genetika untuk menentukan nilai optimumnya, dan dicobakan dua bentuk fungsi *fitness*.

4. PENDEKATAN ALGORITMA GENETIKA DENGAN JARINGAN SYARAF TIRUAN

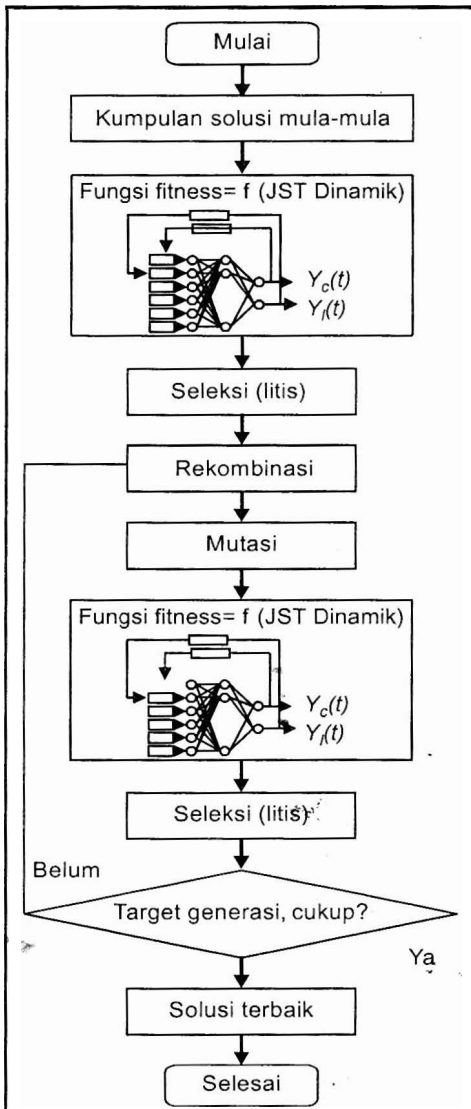
Algoritma genetika tiruan dicirikan dengan kodefikasi kumpulan solusi acak yang disebut populasi. Setiap individu (kromosom) dalam populasi merupakan calon solusi. Individu-individu ini kemudian berevolusi melalui beberapa iterasi atau sering disebut generasi. Pada setiap iterasi, individu dievaluasi menggunakan fungsi fitness. Individu yang nilai fitness lebih tinggi akan terus dipertahankan dan akan mengalami rekombinasi dan mutasi.

Hasil dari rekombinasi dan mutasi adalah individu-individu baru yang

mewarisi sifat solusi dari iterasi sebelumnya. Rekombinasi antara individu yang nilainya fitnessnya lebih baik diharapkan menghasilkan individu yang nilai fitnessnya lebih baik pula. Setelah beberapa iterasi algoritma ini akan menghasilkan individu terbaik yang diharapkan merupakan solusi optimal dari permasalahan yang dihadapi.

Proses-proses dalam algoritma genetika tiruan, seperti pada proses

kodefikasi, rekombinasi, mutasi dan evaluasi mempunyai banyak metode, dan kini semakin berkembang. Algoritma ini biasanya dibangun dalam bentuk perangkat lunak (komputasi) dan pada aplikasinya menggunakan parameter-parameter ukuran populasi, probabilitas rekombinasi (P_c), probabilitas mutasi (P_m) dan banyaknya iterasi. Kesemuanya ditujukan untuk menghasilkan iterasi yang konvergen dan menghindari optimum lokal. Adapun algoritma genetika dengan JST dinamik dapat dilihat pada Gambar 1



Gambar 1. Arsitektur algoritma genetika dengan fungsi fitness JST repon dinamik

Kumpulan solusi mula-mula.

Sekumpulan solusi yang dipresentasikan oleh populasi individu dibangkitkan secara acak dalam bentuk string biner. Setiap individu, untuk kebutuhan ini, memberikan informasi 4 variabel, larutan nutrisi yang diberikan, suhu, kelembaban dan iradiasi. Panjang string biner adalah $4 \times n$, dimana n merupakan jumlah gen tiap variabel, pada implementasi ini $n=8$. Nilai dalam biner ini kemudian dikonversi ke desimal dan dipetakan kedalam nilai minimum dan maksimum setiap variabel. Nilai hasil pemetaan setiap variabel digunakan untuk menghitung nilai fitness total dari model jaringan syaraf tiruan yang telah dikembangkan sebelumnya. Nilai variabel minimum dan maksimum diintroduksi dari hasil pengukuran percobaan lapang.

Fungsi fitness.

Evaluasi yang menggunakan fungsi *fitness* pada prinsipnya membandingkan solusi-solusi (individu-individu) yang tersedia. Pada makalah ini dicobakan dua fungsi *fitness* sebagai berikut:

$$F_1 = \sum_i^n \alpha Y_c(t) + \beta \left(\frac{l}{Y_l(t)+c} \right) \tag{1}$$

$$F_2 = \sum_i^n \alpha Y_c(t) * \left(\frac{l}{\beta Y_l(t)+c} \right) = \sum_i^n \frac{\alpha Y_c(t)}{\beta Y_l(t)+c} \tag{2}$$

dimana adalah fungsi *fitness* 1, adalah fungsi *fitness* 2, α dan β , adalah koefisien

untuk $Y_c(t)$ adalah rasio luas kanopi-diameter batang (rasio kanopi-diameter) pada saat t (cm^2/cm), $Y_i(t)$ adalah limpasan nutrisi (limpasan) pada saat t (ml), c adalah konstanta, i adalah data ke- i , dan n adalah data ke- n .

Persamaan (1) dan (2) disusun dengan tujuan untuk memaksimalkan nilai *fitness* (F), yang pada prinsipnya untuk memaksimalkan rasio kanopi-diameter dan meminimalkan limpasan nutrisi. Jadi disini limpasan nutrisi merupakan faktor pembatas atau pengendali dari faktor rasio kanopi-diameter. Konstanta c sebagai antisipasi untuk menghindari variabel $Y_i(t)$ sama dengan nol, untuk mengakomodasi keperluan tersebut tetapi juga diharapkan nilai c ini tidak berpengaruh signifikan, maka ditentukan nilai c adalah 10^{-6} . Fungsi tujuan yang digunakan adalah jaringan syaraf tiruan (JS) respon dinamik.

Seleksi dan series data.

Seleksi didasarkan pada parameter ukuran populasi (jumlah solusi) mula-mula, artinya jumlah solusi dipertahankan tetap. Banyaknya penambahan solusi sebagai hasil dari rekombinasi, yang banyaknya tergantung nilai parameter ukuran populasi dan probabilitas rekombinasi (P_c), diabaikan. Adapun prosesnya adalah dengan cara menggabungkan terlebih dahulu seluruh individu sebelumnya dengan individu hasil rekombinasi, kemudian diurutkan (*series*) sesuai dengan nilai *fitness*-nya dan diambil sejumlah yang terbaik sesuai dengan parameter ukuran populasi mula-mula. Nilai *fitness* disusun berurut (*series*), yang disimpan dalam *array* 1 sedangkan data populasi (kumpulan solusi) disimpan dalam string *array* 2. *Series data* didasarkan pada perubahan nilai *fitness* yang disimpan di *array* 1 sedangkan string solusi pada *array* 2 hanya menyesuaikan. Dengan cara demikian urutan nilai *fitness* akan bersesuaian dengan urutan solusi.

Rekombinasi

Rekombinasi (*cross over*) dilakukan secara acak pada solusi-solusi terseleksi yang mempunyai nilai fungsi *fitness* tertinggi dalam satu iterasi. Dan jumlahnya yang mengalami rekombinasi didasarkan pada besarnya parameter probabilitas rekombinasi (P_c). Misalnya nilai P_c adalah 0,2, maka diharapkan 20% dari populasi akan mengalami rekombinasi, yaitu 5 dari 25 solusi. Karena untuk melakukan rekombinasi jumlah solusi harus genap, maka jumlah yang ganjil digenapkan ke bawah. Untuk kasus di atas, jumlah solusi yang mengalami rekombinasi adalah empat. Pada Solusi-solusi tersebut dilakukan pemotongan pada empat tempat (karena ada 4 variabel). Rekombinasi hanya terjadi pada variabel variabel yang bersesuaian, misalnya antara variabel suhu pada solusi 1 dengan variabel suhu pada solusi 2, demikian seterusnya.

Mutasi

Mutasi terjadi pada level gen dan dilakukan secara acak dengan cara membangkitkan bilangan acak. Bilangan acak dibangkitkan sebanyak jumlah gen pada populasi. Bila bilangan acak untuk gen pada solusi tersebut lebih kecil dibandingkan dengan parameter probabilitas mutasi (P_m) maka gen itu akan mengalami mutasi, yaitu string 0 akan berubah menjadi 1 atau sebaliknya. Jumlah gen yang mengalami mutasi didasarkan pada parameter P_m . Misalnya nilai P_m adalah 0,01, maka diharapkan 1% dari populasi akan mengalami mutasi, yaitu 8 dari 800 gen. Jumlah 800 gen merupakan hasil dari mengasumsikan jumlah populasi adalah 25 solusi, masing-masing solusi terdiri 4 variabel, dan masing-masing variabel mempunyai panjang 8 gen.

Iterasi.

Iterasi adalah proses pengulangan pada tahap rekombinasi, mutasi, evaluasi

(penghitungan nilai *fitness*, dalam hal ini menggunakan bobot JST respon dinamik), dari seleksi. Penentuan banyaknya iterasi memperhatikan kemungkinan terjebak pada maksimum lokal, artinya walaupun hasil iterasi konvergen tetapi terkondisi pada optimum lokal, tujuan penyelesaian optimasi dengan algoritma genetika tidak akan tercapai.

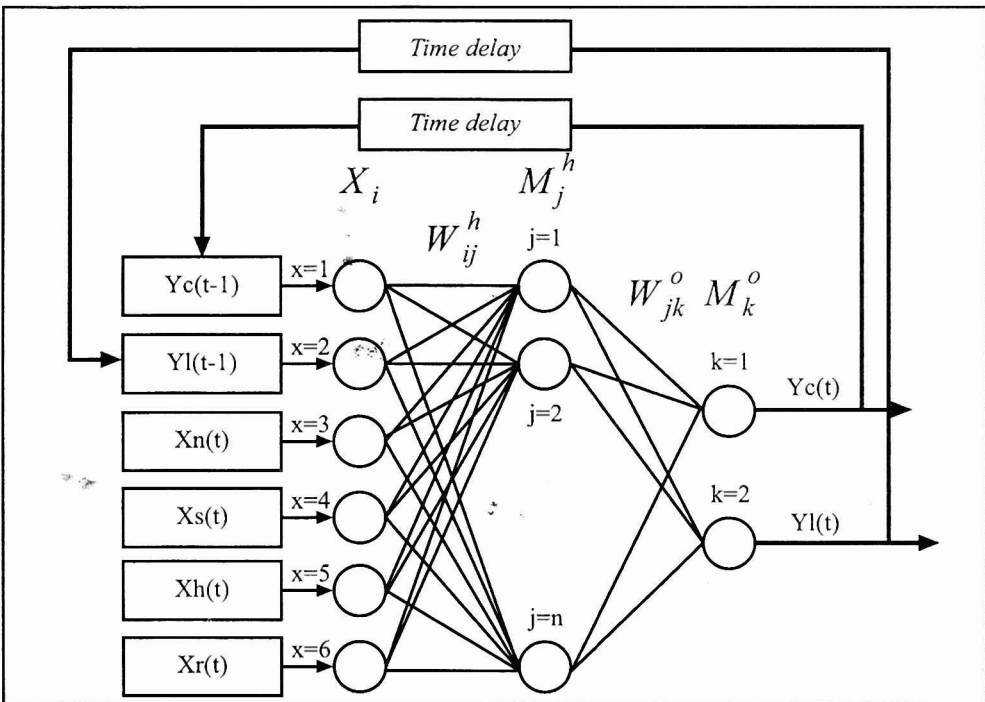
METODE PENELITIAN

Aplikasi teknik optimasi ini menggunakan satu set data hasil pengukuran percobaan dilapangan di Greenhouse PT Saung Mirwan-Bogor, satu set komputer PC dengan spesifikasi CPU Pentium (R) 4, memory RAM 512 MB dan hardisk 40 GB, perangkat lunak sistem operasi Windows XP, dan bahasa pemrograman Delphi 5.0.

Teknik optimasi algoritma genetika dengan fungsi *fitness* jaringan syaraf

tiruan respon dinamik didesain untuk menentukan solusi optimal dari masukan model jaringan syaraf tiruan respon dinamik untuk pertumbuhan tanaman ketimun mini pada fase vegetatif yang telah dikemukakan pada paper sebelumnya (Tamrin *et al.*, 2005). Adapun arsitektur sistem hubungan lingkungan dan tanaman dengan JST respon dinamik tersebut dapat dilihat pada Gambar 2.

Pada algoritma JST penjalaran balik respon dinamik (Gambar 2), awalnya diberikan vektor masukan $X=(1,2, \dots,6)^T$. Kemudian secara berturut-turut dihitung: masukan total dari setiap simpul pada lapisan tersembunyi (pers. 1), keluaran dari setiap simpul pada lapisan tersembunyi (pers. 2), masukan total dari setiap simpul pada lapisan keluaran (pers. 3), keluaran dari setiap simpul pada lapisan keluaran (pers. 4), eror dari setiap simpul pada lapisan keluaran (pers. 5), eror dari setiap simpul pada lapisan tersembunyi (pers. 6), bobot pada lapisan tersembunyi (pers. 7), dan mengubah



Gambar 2. Arsitektur sistem tanaman dengan JST respon dinamik

harga bobot pada lapisan keluaran (pers. 8). Adapun persamaan-persamaan yang dimaksud adalah sebagai berikut,

$$M_j^h = \sum_{i=1}^n W_{ij}^h X_i \quad (1)$$

$$M_j^h = f_j^h(M_j^h) \quad (2)$$

$$M_k^o = \sum_{j=1}^n W_{jk}^o M_j^h \quad (3)$$

$$M_k^o = f^o(M_k^o) \quad (4)$$

$$\delta_k^o = Y_k - M_k^o \quad (5)$$

$$\delta_j^h = (M_j^h) \left(\sum_{k=1}^2 Y_k - M_k^o \right) W_{jk}^o \quad (6)$$

$$W_{ij}^o(t+1) = W_{ij}^o(t) + \eta \delta_j^h X_i \quad (7)$$

$$W_{jk}^o(t+1) = W_{jk}^o(t) + \eta \delta_k^o X_j^h \quad (8)$$

Solusi optimal masukan yang dimaksud adalah solusi optimal masukan setiap hari selama 15 hari (umur fase vegetatif tanaman ketimun mini).

Masukan terdiri dari empat faktor (*decision variable*), yaitu larutan nutrisi (n) yang diberikan, suhu (s), kelembaban (h) dan iradiasi (r). Kendala merupakan nilai minimum-maksimum dari hasil pengukuran setiap hari, selama 15 hari, untuk masing-masing faktor masukan. Sedangkan keluaran adalah dua faktor, rasio luas kanopi-diameter batang ($Y_c(t)$) dan larutan nutrisi yang hilang ($Y_l(t)$).

Pada keluaran, faktor tanaman diupayakan maksimum sedangkan faktor limpasan sebaliknya, yaitu mimimum atau sekecil-kecilnya. Untuk memenuhi prinsip algoritma genetika yang selalu mencari solusi kearah positif (maksimum) maka untuk mencari limpasan yang minimum, faktor limpasan dalam fungsi *fitness* ditulis dalam bentuk resiprokal.

Karena kedua faktor di atas (tanaman dan limpasan) bersifat dualisme, yaitu bertentangan maka alternatif persamaan fungsi *fitnessnya*, diantaranya adalah seperti pada persamaan (1) dan (2) yang dikemukakan sebelumnya. Selanjutnya penggunaan kedua persamaan tersebut pada fungsi *fitness* dalam mencari solusi optimal dieksplorasi.

HASIL DAN PEMBAHASAN

Bobot (weight) model komputasi jaringan syaraf tiruan respon dinamik yang digunakan untuk menghitung fungsi tujuan (*fitness*) dari masukan (nutrisi yang diberikan, suhu, kelembaban dan iradiasi) terhadap keluaran rasio kanopi-diameter dan limpasan, untuk setiap hari, selama 15 hari, dapat dilihat pada Gambar 2. Selanjutnya, optimasi keluaran tersebut dieksplorasi dengan menggunakan algoritma genetika (Gambar 1).

Pada Gambar 4 dan 5 terlihat kurva evolusi untuk mencari nilai optimal dengan parameter-parameter yang berbeda, secara berturut-turut menggunakan pers (1) dan pers (2). Kedua persamaan ini dilihat konsistensinya dalam memenuhi prioritas yang diinginkan, yaitu memprioritaskan peningkatan rasio kanopi-diameter, $Y_c(t)$ daripada penurunan limpasan, $Y_l(t)$. Sedangkan seberapa besar peran yang diharapkan dari komponen $Y_c(t)$ dan $Y_l(t)$ di dalam persamaan fungsi *fitness*, dilakukan dengan mengatur koefisien α dan β yang disesuaikan dengan kebutuhan, dalam kasus ini α dan β adalah 0,5.

Pada kedua kurva evolusi untuk keempat skenario terlihat bahwa kisaran (range) nilai *fitness* yang dihasilkan untuk pers (1) adalah sekitar 2650 - 9700 (Gambar 4) dan untuk pers (2) adalah 1160 - 15759380 (Gambar 5). Dengan kisaran masing-masing demikian, maka rentang (selisih batas atas dan bawah

kisaran) nilai *fitness* yang dihasilkan untuk pers(1) adalah 7050 dan untuk pers (2) adalah 15758220. Nilai rentang yang lebih kecil pada pers (1), menunjukkan penelusuran optimum *global* dapat lebih cepat. Sedangkan nilai rentang yang lebih besar pada pers (2) cenderung untuk terjebak dalam optimum lokal, dan hal ini terlihat jelas setelah mensimulasikan kembali hasil optimal masing-masing masukan ke dalam model JST.

Pada Gambar 7 dan 9 adalah hasil simulasi dari masukan yang dihasilkan skenario ($P_c=0,6$; $P_m=0,01$) dan ($P_c=0,6$; $P_m=0,1$) kurva pers (1) dari Gambar 4. Sedangkan Gambar 11 dan 13 adalah berturut-turut skenario yang sama untuk kurva pers (2) dari Gambar 5. Terlihat bahwa keluaran, yaitu rasio kanopi-diameter dan limpasan dari kedua skenario untuk pers (2) dihasilkan dari masukan dengan penelusuran nilai *fitness* optimum lokal, demikian juga skenario ($P_c=0,6$; $P_m=0,1$) untuk pers (1). Sedangkan skenario ($P_c=0,6$; $P_m=0,01$) untuk pers (1) dihasilkan dari masukan dengan penelusuran nilai *fitness* optimum *global*.

Perbedaan signifikan kedua parameter, kisaran dan rentang, antara pers (1) dan (2) dikarenakan penyusunan komponen $Y_c(t)$ dan $Y_l(t)$ dalam kedua persamaan tersebut sedemikian rupa. Secara teoritis, pers (2), karena akan menghasilkan nilai *fitness* yang jauh lebih tinggi, diharapkan berimplikasi kepada pencapaian optimum global lebih cepat, namun hal ini tidak terjadi.

Pada pers (2) tidak terjadi konsistensi dalam memprioritaskan $Y_c(t)$ daripada $Y_l(t)$. Pada tabel 1 terlihat bahwa untuk kasus 1, 2, dan 3, sesuai dengan prioritas maka pilihan terbaik adalah kasus 1. Hal ini konsisten ditunjukkan oleh dengan nilai yang tertinggi pada kasus 1, tetapi tidak demikian dengan . Hal serupa terjadi pada kasus 4, 5, dan 6, sesuai dengan prioritas maka pilihan terbaik adalah kasus 6. Hal ini konsisten ditunjukkan

Tabel 1. Konsistensi prioritas

Kasus	$Y_c(t)$	$Y_l(t)$	F_1	F_2
1	50	300	25,00167	15000
2	50	310	25,00161	15500
3	50	320	25,00156	16000
4	100	3	50,16667	300
5	200	2	100,25000	400
6	201	1	101,00000	201

oleh dengan nilai tertinggi pada kasus 6, tetapi tidak demikian dengan.

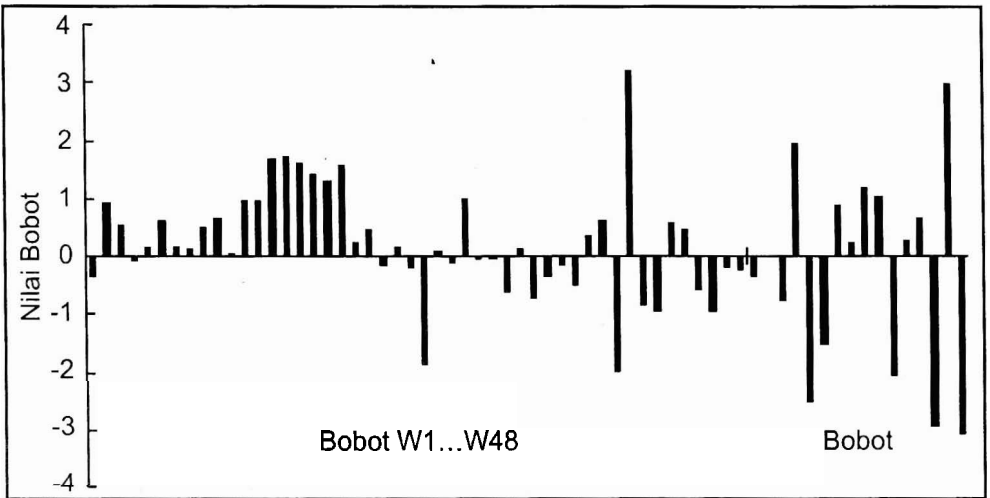
Kondisi optimum *global*, bila hanya dilihat dari kurva evolusi penelusuran *fitness* pers (1) dan (2), masing-masing pada Gambar 4 dan 5, performa perbedaannya tidak terlihat jelas. Keduanya sama-sama konvergen bahkan performa untuk pers (2) tampak lebih baik karena nilai *fitness* pers (2) lebih tinggi. Hal ini dapat dilihat untuk skenario ($P_c=0,6$; $P_m=0,01$) pada Gambar 4-pers (1) dan Gambar 5-pers (2).

Untuk melihat perbedaan performa penggunaan pers (1) dan (2) yang lebih jelas, yaitu dengan mensimulasikan kembali hasil masing-masing skenario pada penelusuran (*searching*) optimal untuk kedua persamaan tersebut ke model JST respon dinamik yang digunakan sebagai fungsi *fitness*.

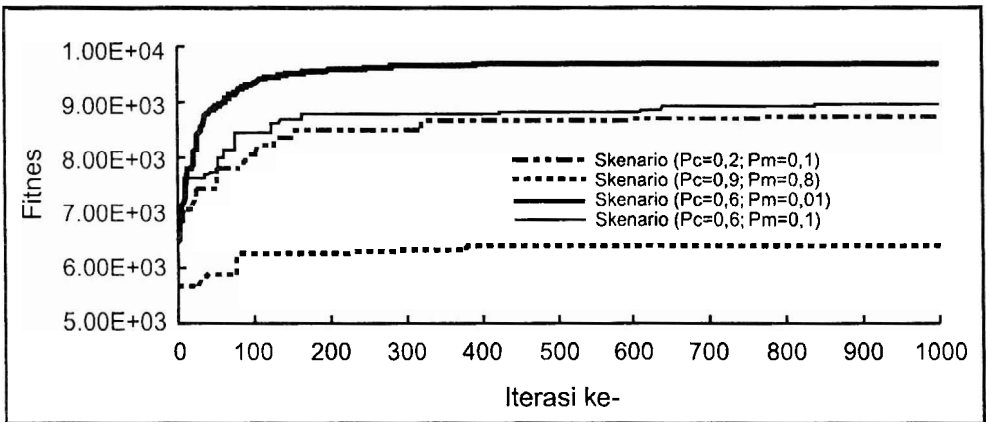
Hasil optimal penelusuran kurva pers (1) skenario ($P_c=0,6$; $P_m=0,01$) sebagai masukan dan keluaran model JST, masing-masing dapat dilihat pada Gambar 6 dan 7, sedangkan skenario ($P_c=0,6$; $P_m=0,1$) dapat dilihat pada Gambar 8 dan 9.

Sedangkan hasil optimal kurva pers (2) skenario ($P_c=0,6$; $P_m=0,01$) sebagai masukan dan keluaran model JST, masing-masing dapat dilihat pada Gambar 10 dan 11. Untuk skenario ($P_c=0,6$; $P_m=0,1$) dapat dilihat pada Gambar 12 dan 13.

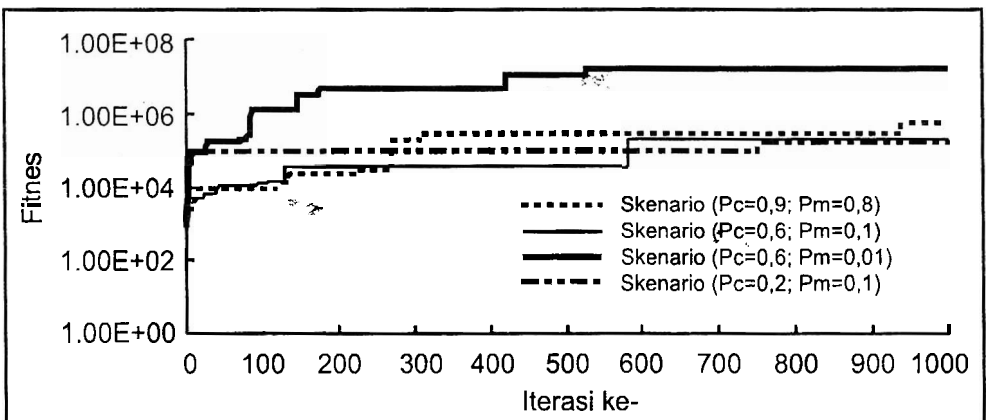
Skenario yang menunjukkan hasil stabil, yaitu berada paling atas baik pada kurva evolusi penelusuran *fitness* dengan



Gambar 3. Bobot model komputasi JST respon dinamik



Gambar 4. Kurva evolusi penelusuran fitness dengan pers (1)



Gambar 5. Kurva evolusi penelusuran fitness dengan pers (2)

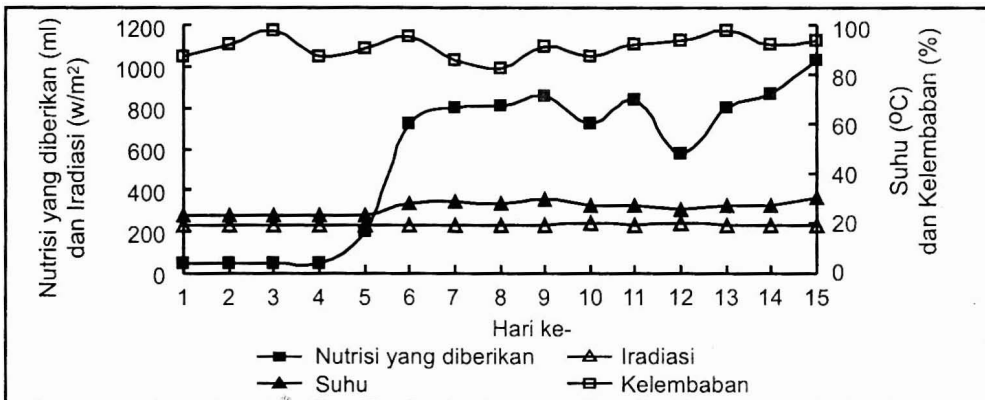
pers (1) dan pers (2), adalah skenario ($P_c=0,6$; $P_m=0,01$). Namun demikian, yang merupakan hasil terbaik, yaitu pencapaian optimum global adalah skenario ($P_c=0,6$; $P_m=0,01$ dengan pers (1). Hal ini dapat dilihat pada hasil simulasi model JST yang menggunakan masukan hasil penelusuran kedua persamaan tersebut untuk skenario yang sama, yaitu masing-masing dapat dilihat pada Gambar 7 dan 13. Hasil pertumbuhan tanaman (rasio kanopi-diameter) pada Gambar 7 lebih konsisten daripada yang ditunjukkan oleh Gambar 13.

Penjelasan-penjelasan di atas menunjukkan penelusuran optimum global dengan pers (1) konsisten terhadap prioritas yang diharapkan. Sedangkan perilaku skenario yang dicobakan pada

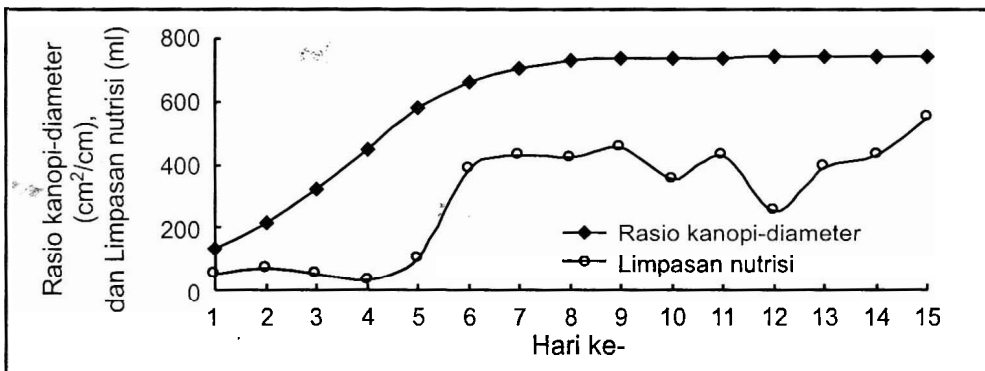
pers (1) dijelaskan berikut ini.

Skenario ($P_c=0,6$; $P_m=0,01$) pada Gambar 4, *fitness* mula-mula naik dengan gradien cukup tajam sampai iterasi ke-100, kemudian secara perlahan gradien kenaikan berkurang dan akhirnya mencapai nilai maksimum pada iterasi sekitar 400. Biasanya iterasi penelusuran dihentikan bila nilai maksimum *fitness* tetap sama dengan bertambahnya iterasi dan solusi optimal diberikan oleh individu dengan nilai maksimum tersebut.

Untuk menghindari optimum lokal, iterasi penelusuran diteruskan sampai iterasi ke-1000. Dan untuk skenario ($P_c=0,6$; $P_m=0,01$) tetap stabil, sedangkan skenario lainnya tidak demikian. Perilaku yang mirip dengan skenario ($P_c=0,6$; $P_m=0,01$) adalah skenario ($P_c=0,9$;



Gambar 6. Hasil optimal dari penelusuran optimum global kurva pers (1) skenario ($P_c=0,6$; $P_m=0,01$)



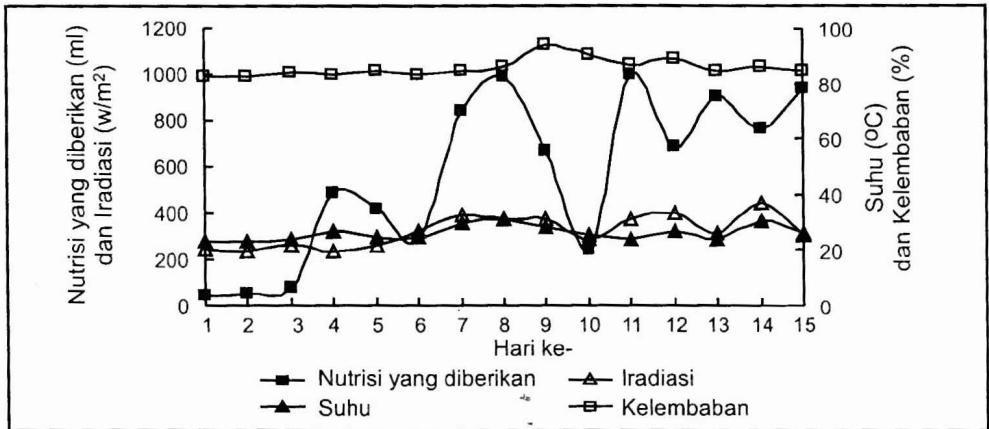
Gambar 7. Hasil simulasi model JST dengan masukan hasil optimum global penelusuran kurva pers (1) skenario ($P_c=0,6$; $P_m=0,01$)

$P_m=0,8$) tetapi nilai maksimum yang dicapai nya dibawah nilai skenario ($P_c=0,6$; $P_m=0,01$) atau yang lainnya. Hal ini mengindikasikan penggunaan parameter $P_c=0,9$ dan $P_m=0,8$ terlalu besar untuk permasalahan ini.

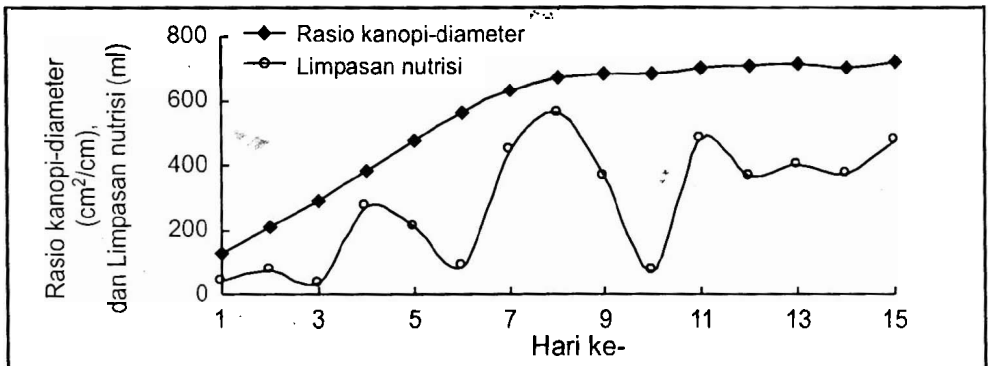
Metode seleksi yang digunakan adalah *elitist*. Menurut Morimoto *et al.* (2001), dan Morimoto dan Hashimoto (1998), kelebihan metode seleksi ini adalah efektif untuk meningkatkan *fitness* individu karena individu dengan nilai *fitness* lebih tinggi dipertahankan untuk iterasi berikutnya, dan hal ini sekaligus menjadi kelemahannya karena akan mudah terperangkap pada kondisi optimum lokal.

Pada studi ini untuk mengatasi hal ini digunakan jumlah populasi (*pop size*)

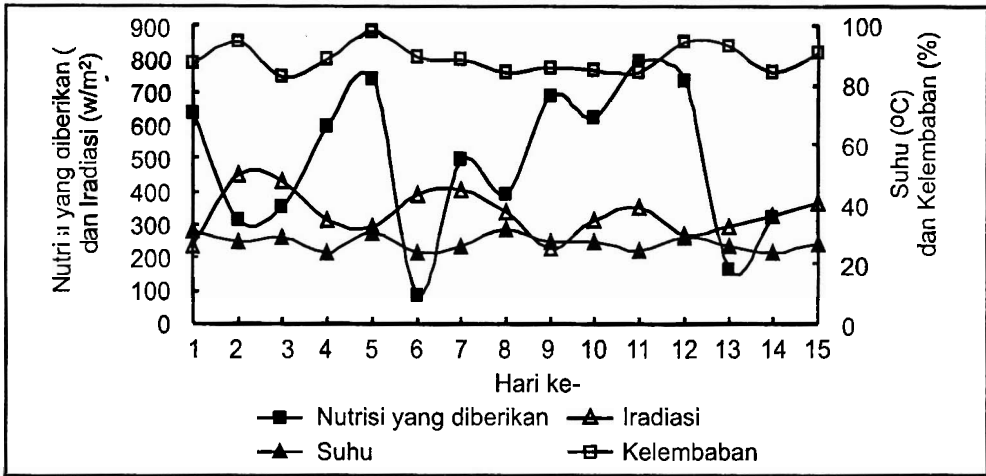
yang cukup besar, yaitu 35. Berdasarkan performa optimum global yang dijelaskan sebelumnya, maka jumlah populasi yang digunakan telah memadai, selain itu Kusumadewi (2003) mengemukakan bahwa jumlah populasi sebaiknya tidak kurang dari 30. Otimum global dengan penelusuran algoritma genetika tidak selalu tercapai. Untuk konfirmasi, pada makalah ini menggunakan metode cek-silang, yaitu dengan cara mensimulasikan kembali solusi hasil penelusuran dengan model JST respon dinamik yang digunakan sebagai fungsi *fitness*. Dalam penggunaan teknik optimasi algoritma genetika, konfirmasi cek silang, diharapkan paling tidak dapat memperkuat dugaan bahwa hasil



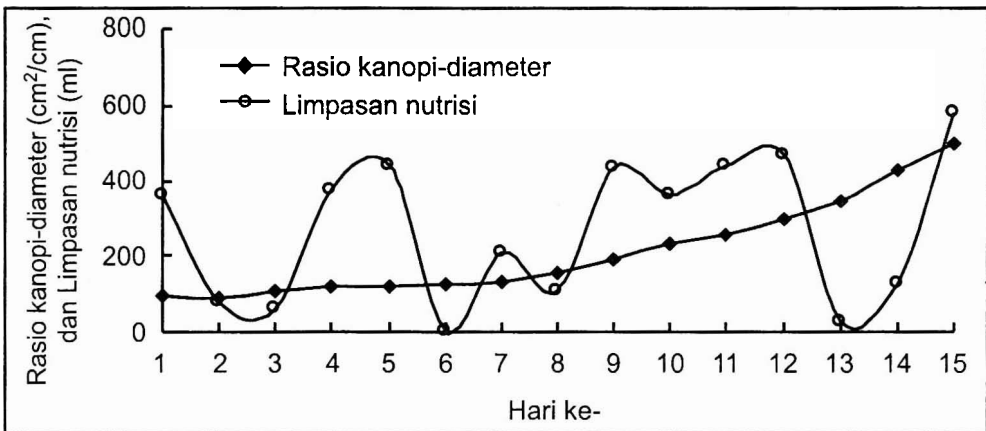
Gambar 8. Hasil dari penelusuran optimum lokal kurva pers (1) skenario ($P_c= 0,6$; $P_m= 0,1$)



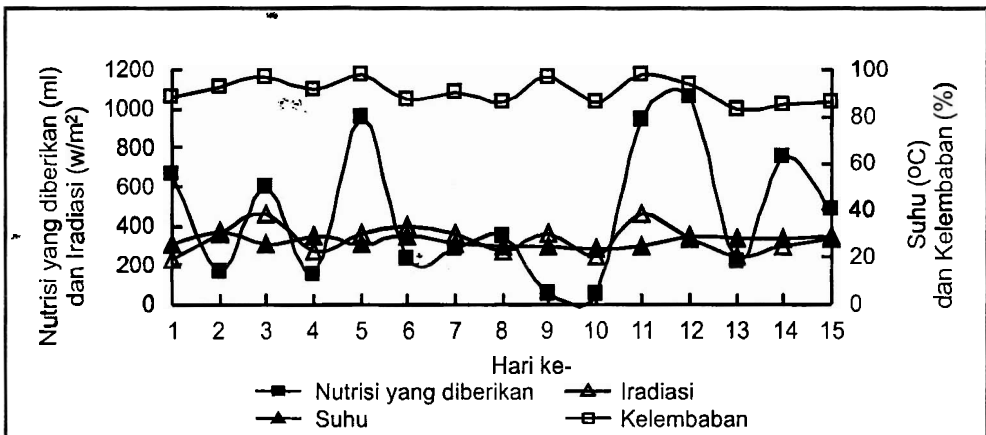
Gambar 9



Gambar 10. Hasil dari penelusuran kurva pers (2) skenario ($P_c=0,6$; $P_m=0,01$)



Gambar 11. Hasil simulasi model JST dengan masukan hasil penelusuran kurva pers (2) skenario ($P_c=0,6$; $P_m=0,01$)



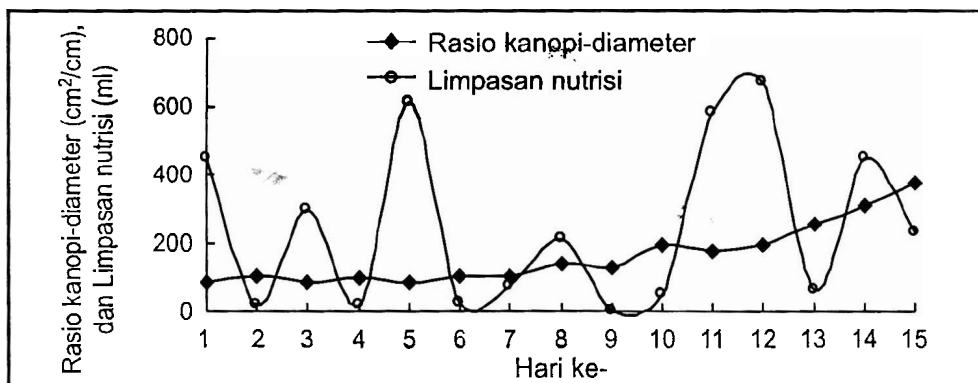
Gambar 12. Hasil dari penelusuran kurva pers (2) skenario ($P_c=0,6$; $P_m=0,1$)

KESIMPULAN

Penggunaan fungsi *fitness* pers (1) dalam mencari solusi optimal untuk masukan (larutan nutrisi yang diberikan, suhu, kelembaban dan iradiasi) model JST respon dinamik guna mendapatkan keluaran maksimum untuk rasio luas kanopi-diameter batang (pertumbuhan vegetatif tanaman) dan keluaran minimum untuk larutan nutrisi yang hilang (limpasan) telah menunjukkan hasil yang konsisten. Skenario ($P_c = 0.6$ and $P_m = 0.01$) menghasilkan optimum global. Adapun konfirmasi optimum global yang digunakan adalah cek-silang, yaitu dengan cara mensimulasikan kembali solusi hasil penelusuran ke dalam model JST respon dinamik yang digunakan sebagai fungsi *fitness*.

DAFTAR PUSTAKA

- Esmay L M, Dixon, J. E. 1986. Environmental Control for Agricultural Building. The Avi Publishing Company, Inc., Westport Connecticut, USA.
- Kusumadewi S. 2003. *Artificial Intelligence (Teknik dan Aplikasinya)*. Yogyakarta: Graha Ilmu.
- Morimoto T, Kato R, Aono T, Hashimoto Y. 2001. Identification and optimal control of the water loss of fruit during storage, as affected by temperature. Di dalam: Purwadaria HK, Widodo RJ, Seminar KB, Tjokronegoro HA, Suroso, editor. Preprints 2nd IFAC-CIGR Workshop on Intelligent Control for Agricultural Applications; Bali; 22-24 Aug 2001. Indonesia: IFAC-CGR. hlm 162-167.
- Morimoto T, Hashimoto Y. 1998. AI approaches to identification and control of total plant production. Di dalam: Kozai T, Murase H, Hosyi T, editor. Preprints 3rd IFAC-CIGR Workshop on Artificial Intelligent in Agriculture; Chiba; 24-26 Apr 1998. Japan: IFAC-CGR. hlm 1-19.
- Tamrin, Seminar KB, Suhardiyanto H, Hardjoamidjojo S. 2005. Model Jaringan Syaraf Tiruan untuk Pertumbuhan Tanaman Ketimun Mini (*Cucumis sativus L. Var. Marla*) pada Fase Vegetatif. Jurnal Keteknik Pertanian. 19(1): 1-10.
- Ursem RK, Filipic B, Thiemo. 2002. Exploring The Performance of an Evolutionary Algorithm for Greenhouse Control. [http://66.102.104/9Search?q=catch:WiPMQJ4rzCsJ:wwww.evalife.dk/publications/RKU ITI2](http://66.102.104/9Search?q=catch:WiPMQJ4rzCsJ:wwww.evalife.dk/publications/RKU%20ITI2). [13 April 2004].
- Yandra, Hermawan A. 2000. Algoritma Genetika dan Aplikasinya untuk Agribisnis Masa Depan. Agrimedia. 6(1): 42-43.



Gambar 13. Hasil simulasi model JST dengan masukan hasil penelusuran kurva pers (2) skenario ($P_c=0,6$; $P_m=0,1$)