

Optimasi Gerakan Mesin Bor Otomatis dengan Menggunakan Algoritma Genetika

Thiang, Anies Hannawati

Dosen Fakultas Teknologi Industri, Jurusan Teknik Elektro, Universitas Kristen Petra
email: thiang@petra.ac.id, anies@petra.ac.id

Tendra Rustan

Alumni Fakultas Teknologi Industri, Jurusan Teknik Elektro, Universitas Kristen Petra

Abstrak

Makalah ini akan menjelaskan tentang mesin bor otomatis menggunakan kamera sebagai sensor untuk mendeteksi koordinat pad dan via dalam sebuah PCB secara otomatis. Beberapa teknologi pemrosesan image digunakan untuk mendeteksi koordinat pad dan via, antara lain threshold, grayscale, fillrect. Algoritma genetika diterapkan untuk mengoptimasi gerakan dari mesin bor sehingga mesin bor dapat melakukan proses pengeboran dengan efektif. Algoritma genetika akan mencari rute yang optimal yang terdiri atas urutan pengeboran pad dan via. Operator genetika yang digunakan dalam system ini adalah seleksi, crossover dan inversi.

Hasil pengujian menunjukkan bahwa sistem dapat berjalan dengan baik dan dapat mendeteksi semua pad dan via dalam PCB. Dengan algoritma genetika, sistem dapat mereduksi waktu pengeboran sampai 50%.

Kata kunci: mesin bor otomatis, algoritma genetika, pemrosesan image, optimasi, PCB.

Abstract

This paper will describe about automatic drilling machine using camera as the sensor in order to detect the coordinates of pad and via automatically. Several image processing technologies were used for detecting the coordinates of pad and via. They are threshold, grayscale, fillrect. Genetic algorithm was used to optimize movement of the drilling machine, so that, the machine can do the drilling process effectively. Genetic algorithm will find the optimal route which consists of the drilling sequence of pad and via. Genetic operators used in this system are selection, crossover and inversion.

Experiments were done. Experiment result showed that the system could run well and detect all holes in PCB. The system achieved a time reduction rate up to 50 %.

Keywords: mesin bor otomatis, algoritma genetika, pemrosesan image, optimasi, PCB.

1. Pendahuluan

Penggunaan Printed Circuit Board (PCB) merupakan hal yang penting jika ingin membangun suatu rangkaian elektronika yang baik. Pada PCB terdapat *pad* dan *via* dimana keduanya harus dibor agar komponen elektronika dapat dipasang pada PCB. Untuk rangkaian yang kompleks, jumlah *pad* dan *via* sangat banyak dan ini dapat menimbulkan kesalahan pada saat proses pengeboran.

Makalah ini akan membahas tentang otomatisasi mesin bor dengan menggunakan kamera

untuk mendeteksi koordinat pengeboran secara otomatis. Kamera berfungsi menangkap gambar print out dari PCB kemudian dengan metode pemrosesan image dapat diketahui dan diambil koordinat pengeborannya. Metode pemrosesan image yang digunakan untuk pendeteksian koordinat antara lain threshold, gray scale dan floodfill. Hasil pengambilan koordinat dalam satuan pixel akan dikonversikan ke satuan milimeter agar dapat dilakukan pengeboran.

Namun mesin bor otomatis yang telah dibuat itu memiliki kelemahan di mana gerakan mesin bor tersebut masih kurang efektif. Oleh karena digunakan algoritma genetika untuk mengefektifkan gerakan mesin bor.

Selanjutnya makalah ini akan diorganisasi sebagai berikut: bagian kedua akan dijelaskan

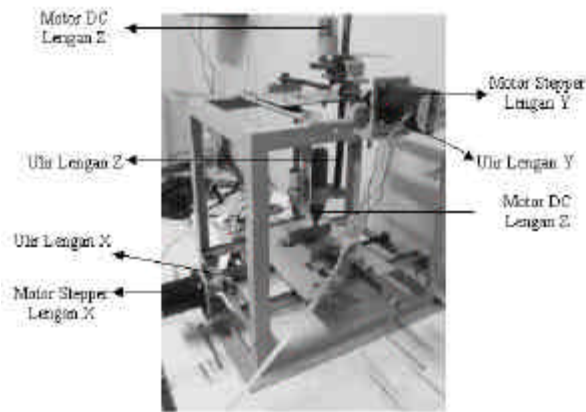
Catatan : Diskusi untuk makalah ini diterima sebelum tanggal 1 Februari 2004. Diskusi yang layak muat akan diterbitkan pada Jurnal Teknik Mesin Volume 6 Nomor 1 April 2004.

secara singkat mesin bor otomatis yang ada. Berikutnya pada bagian ketiga akan dijelaskan tentang proses pendeteksian koordinat bor. Pada bagian keempat akan dijelaskan tentang penerapan algoritma genetika, diikuti dengan pengujian sistem pada bagian kelima. Terakhir akan disimpulkan hal-hal yang berkaitan dengan proyek penelitian ini.

2. Mesin Bor Otomatis

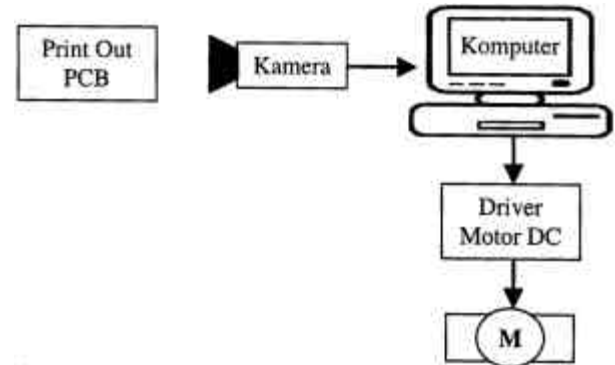
Pada bagian ini akan dijelaskan secara umum mengenai mesin bor otomatis yang telah dibuat. Gambar 1 menunjukkan model mekanik mesin bor yang telah didisain. Komponen mesin bor otomatis seperti yang terlihat pada gambar terdiri atas 2 buah motor stepper untuk lengan X dan Y, sebuah motor DC untuk lengan Z, dan sebuah motor DC sebagai bor PCB. Pergerakan mesin bor otomatis ini pada setiap sumbu nya dirancang mempergunakan ulir. Penggunaan ulir ini bertujuan agar pergeseran lengan akan lebih teliti. Jarak antar ulir yang digunakan sebesar 1,588 mm, maka untuk putaran motor stepper sebesar 360⁰ (1 putaran) akan didapatkan pergeseran lengan sebesar 1,588 mm.

Motor stepper baik untuk lengan X maupun Y memiliki 200 langkah dalam satu putaran (360⁰). Sehingga untuk satu langkah didapatkan sudut sebesar 1,8⁰. Seperti yang diketahui bahwa jarak antar ulir adalah 1,588 mm untuk 1 putaran dan dalam 1 putaran ada 200 langkah. Jadi jarak pergeseran untuk 1 langkah (1,8⁰) adalah sebesar 0.00794 mm.



Gambar 1. Model Mekanik Mesin Bor

Semua proses pendeteksian koordinat bor dan proses kontrol mesin bor yang telah didisain dilakukan oleh sebuah komputer yang dilengkapi dengan sebuah kamera. Gambar 2 menunjukkan blok diagram perangkat keras dari mesin bor yang telah didisain.



Gambar 2. Blok Diagram Perangkat Keras Mesin Bor

3. Proses Pendeteksian Koordinat Pengeboran

Pendeteksian koordinat bor pad dan via dilakukan dengan memproses image yang diperoleh dari kamera. Ukuran image yang digunakan adalah 320 x 240 pixel. Beberapa batasan gambar print out PCB yang perlu diperhatikan agar dapat diproses untuk pendeteksian koordinat bor adalah sebagai berikut:

- Print out PCB berwarna hitam putih dengan ukuran maksimum 9 x 10 cm.
- Layout PCB dikelilingi oleh frame berbentuk kotak dan berwarna hitam.
- Pada layout pad dan via terdapat lubang titik pengeboran.

Gambar 3 menunjukkan blok diagram sistem pendeteksian koordinat pengeboran. Secara umum, proses pendeteksian koordinat bor terdiri atas empat tahap yaitu pemrosesan awal image, proses pengisian image, proses perbaikan image dan perhitungan koordinat bor.



Gambar 3. Blok Diagram Sistem Pendeteksian Koordinat Pengeboran

Tahap pemrosesan awal image terdiri atas dua proses yaitu pengubahan format image dari RGB menjadi grayscale dan proses threshold. Pengubahan format image dari RGB menjadi grayscale dilakukan dengan menggunakan metode gray illuminance. Metode ini direpresentasikan dengan menggunakan persamaan berikut:

Gray = 0,299R + 0,587G + 0,114B (1)

Proses threshold dilakukan dengan menggunakan persamaan berikut:

$$g(x,y) = \begin{cases} 255 & \text{jika } f(x,y) \geq T \\ 0 & \text{untuk yang lain} \end{cases} \quad (2)$$

dimana T adalah nilai threshold. Hasil dari proses ini adalah image dua warna yaitu warna hitam dan putih.

Setelah tahap pemrosesan awal image, dilakukan proses pengisian image. Dalam proses pengisian image ini, semua warna putih kecuali lubang bor yang ada pada pad dan via akan diisi dengan warna hitam. Hasil dari proses ini adalah seluruh image akan berwarna hitam kecuali lubang-lubang bor pada pad dan via.

Pada tahap berikut, akan dilakukan sekali lagi proses threshold dengan tujuan membalik warna hitam menjadi putih dan putih menjadi hitam. Sehingga dalam image, seolah-olah hanya ada titik-titik berwarna hitam yang merupakan titik-titik dimana PCB harus dibor.

Tahap terakhir adalah menghitung koordinat dari titik-titik pengeboran. Hal ini dilakukan dengan mencari batas atas (Y_{MAX}), batas bawah (Y_{MIN}), batas kiri (X_{MIN}) dan batas kanan (X_{MAX}) dari pad atau via yang akan dibor. Perhitungan koordinat bor dilakukan dengan menggunakan persamaan berikut:

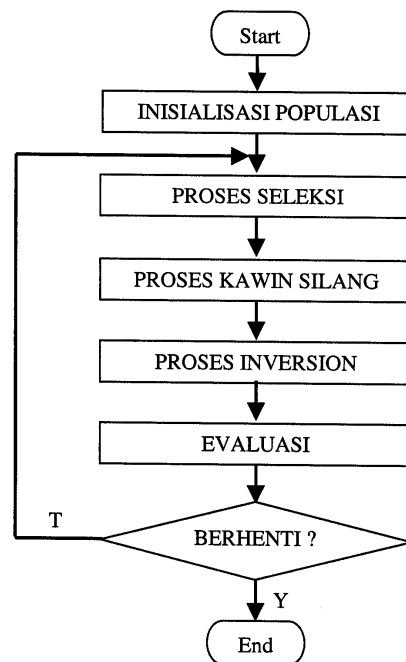
$$X = \frac{X_{MIN} + X_{MAX}}{2} \quad (3)$$

$$Y = \frac{Y_{MIN} + Y_{MAX}}{2} \quad (4)$$

4. Implementasi Algoritma Genetika

Untuk mengefisienkan gerakan pengeboran dari mesin bor, pada sistem yang telah didisain, diterapkan algoritma genetika. Tujuan algoritma genetika adalah mencari rute urutan pengeboran yang paling optimum. Kriteria rute optimum disini adalah rute dengan jarak tempuh paling pendek. Tentunya dalam rute tersebut, setiap pad atau via dari PCB yang akan dibor hanya boleh dilewati satu kali saja.

Skema algoritma genetika dapat dilihat pada Gambar 4. Secara garis besar proses algoritma genetika dimulai dengan inisialisasi populasi yang merupakan inisialisasi sekumpulan alternatif pemecahan yang didapat secara acak. Kemudian dari populasi awal ini akan dibentuk populasi generasi baru dengan menggunakan operator algoritma genetika. Demikian seterusnya proses pembentukan generasi baru dilakukan berulang-ulang hingga kriteria berhenti telah dicapai. Operator algoritma genetika yang digunakan dalam sistem ini adalah seleksi, kawin silang dan inversion.



Gambar 4. Skema Algoritma Genetika

1. Fungsi Fitness dan Representasi Kromosom

Karena dalam sistem ini yang ingin dicari adalah rute terpendek, maka fungsi fitness yang digunakan adalah sebagai berikut:

$$f(x) = \sum_{i=1}^j X_i \quad (5)$$

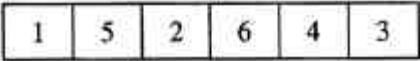
dimana $f(x)$ adalah fungsi fitness, j adalah jumlah pad dan via yang akan dibor dan X adalah jarak antar pad atau via yang akan dibor. Jarak antar pad atau via dapat dihitung dengan menggunakan persamaan berikut:

$$X = \sqrt{(X_1 - X_2)^2 + (Y_1 - Y_2)^2} \quad (6)$$

Representasi kromosom yang digunakan dalam sistem algoritma genetika adalah kromosom berbentuk integer. Kromosom-kromosom berisikan urutan nomor-nomor pad atau via yang akan dibor. Ada beberapa syarat yang harus dipenuhi oleh sebuah kromosom yaitu:

- Nomor pad atau via dari PCB tidak boleh berulang, karena mesin bor otomatis hanya boleh mengebor dan melewati pad atau via tersebut satu kali saja.
- Nomor pad atau via tidak boleh lebih besar dari jumlah keseluruhan pad dan via PCB yang dibor.
- Nomor pad atau via juga tidak boleh nol, karena penomoran pad atau via dimulai dari satu.

Panjang kromosom sangat bergantung pada jumlah pad dan via yang akan dibor. Bila ada 20 pad dan via yang harus dibor, maka kromosom akan mempunyai 20 gen. Gambar 5 menunjukkan contoh representasi kromosom yang menunjukkan urutan pengeboran dari pad atau via 1, 5, 2, 6, 4, 3.



Gambar 5. Contoh Kromosom

2. Seleksi

Ada dua jenis operator seleksi yang digunakan yaitu seleksi roulette wheel dan seleksi tournament. Seleksi roulette wheel bekerja berdasarkan nilai fitness yang dimiliki oleh kromosom-kromosom. Semakin besar nilai fitness yang dimiliki oleh sebuah kromosom maka semakin besar peluang kromosom tersebut untuk terpilih. Sedangkan seleksi tournament bekerja dengan memilih sepasang kromosom secara acak dan dari sepasang kromosom tersebut, kromosom dengan nilai fitness terbesar yang akan terpilih.

Karena dalam aplikasi ini, algoritma genetika digunakan untuk mencari rute terpendek dan hal ini tidak sesuai dengan prinsip dasar seleksi roulette wheel dan tournament, maka nilai fitness dari kromosom akan diubah dulu dengan menggunakan persamaan berikut:

f'(x) = f(x)_{MAX} + f(x)_{MIN} - f(x_i) (7)

Dengan menggunakan persamaan diatas, maka kromosom semula yang memiliki fitness terbesar akan menjadi kromosom dengan nilai fitness terkecil sehingga peluangnya menjadi kecil untuk terpilih. Nilai fitness baru ini hanya digunakan dalam proses seleksi.

3. Kawin Silang

Metode kawin silang yang digunakan adalah Partially Matched Crossover (PMX). Pemilihan metode ini dikarenakan oleh representasi kromosom yang berbentuk integer. Proses kawin silang ini dimulai dengan memilih dua kromosom yang akan dikawin silang kemudian dilanjutkan dengan menentukan dua titik potong secara acak sesuai dengan panjang kromosom. Setelah ditentukan kedua titik potong tersebut maka proses PMX melakukan penukaran posisi gen kromosom yang berada diantara kedua titik potong tersebut. Proses PMX akan menghasilkan dua buah kromosom baru. Berikut adalah

contoh proses kawin silang PMX antara kromosom A dan kromosom B:



4. Inversion

Proses inversion dilakukan setelah proses kawin silang. Pemilihan metode ini dikarenakan oleh representasi kromosom yang berbentuk integer. Proses inversion dimulai dengan menentukan kromosom yang akan diproses kemudian dilanjutkan dengan menentukan dua titik potong secara acak. Proses inversion dilakukan dengan menukar posisi gen dalam kromosom tersebut. Berikut adalah contoh proses inversion untuk kromosom A.

A = 1 2 | 3 4 5 6 | 7 8
A' = 1 2 | 6 5 4 3 | 7 8

5. Evaluasi Pembentukan Generasi Baru dan Kriteria Berhenti

Dalam system ini, strategi yang digunakan untuk membentuk suatu generasi baru adalah Steady-State-No-Duplicate. Jadi semua kromosom pada generasi lama ditambah dengan offspring hasil kawin silang dan inversion diseleksi untuk membentuk suatu generasi baru. Pemilihan dilakukan dengan memilih kromosom-kromosom yang memiliki nilai fitness terbaik dan dengan syarat tidak boleh ada kromosom yang kembar dalam generasi baru yang terbentuk.

Kriteria berhenti yang digunakan pada sistem algoritma genetika adalah dilakukan pengecekan sebanyak n generasi ke depan, apakah ada kromosom yang lebih baik dari kromosom terbaik saat ini. Semakin banyak generasi yang diperiksa maka akan semakin banyak waktu yang dibutuhkan untuk proses algoritma genetika.

5. Pengujian

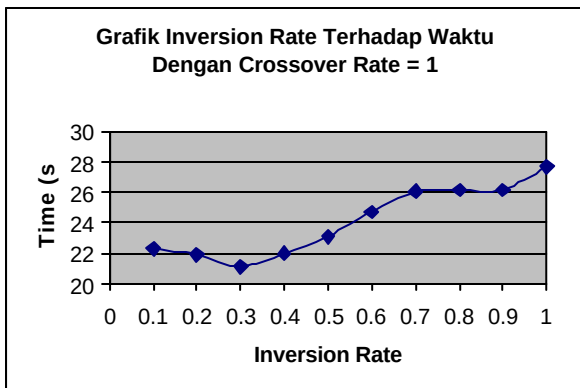
Pada bagian ini akan dibahas pengujian pada proses algoritma genetika dan sistem secara keseluruhan. Pengujian dilakukan dengan menggunakan 3 buah PCB yang memiliki jumlah hole yang berbeda-beda. PCB pertama memiliki jumlah pad dan via 16 buah yang terdiri dari header 5 x 2, dan conector 6 pin. PCB kedua memiliki jumlah hole 21 buah yang terdiri dari sebuah DB-9, resistor, dan IC 4

pin. Dan PCB ketiga memiliki jumlah *hole* 34 buah yang terdiri dari IC 20 pin, resistor dan transistor.

Ada beberapa pengujian yang akan dilakukan, yaitu pengujian dengan variasi nilai crossover rate dan inversion rate, pengujian dengan variasi population size, pengujian terhadap kriteria penghentian generasi, pengujian variasi metode seleksi, pengujian terhadap gerakan mesin bor otomatis.

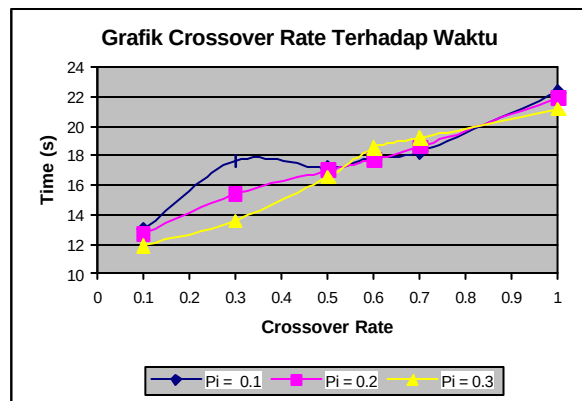
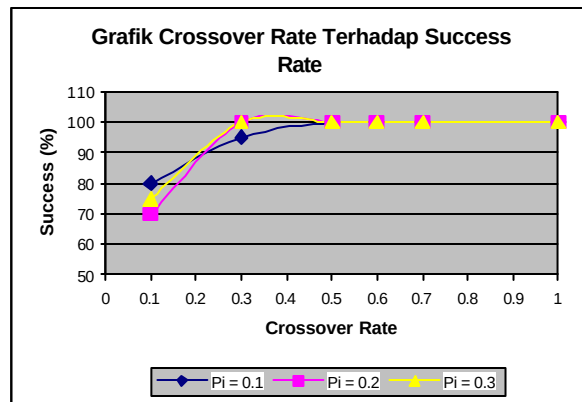
1. Pengujian inversion rate dan crossover rate

Pengujian ini dilakukan untuk menentukan nilai crossover rate dan inversion rate yang terbaik. Dalam hal ini, nilai inversion rate dan crossover rate yang bagus akan menghasilkan suatu solusi dalam waktu yang relatif cepat dengan tingkat keberhasilan yang tinggi. Kemudian nilai inversion rate dan crossover rate yang didapat melalui pengujian ini akan terus digunakan dalam melakukan pengujian selanjutnya. Berikut adalah grafik hasil pengujian variasi nilai inversion rate terhadap waktu proses.



Gambar 6. Grafik Pengujian dengan Variasi Inversion Rate

Dari hasil pengujian terlihat bahwa nilai inversion rate antara 0.1 – 0.4 memiliki waktu proses lebih cepat dibandingkan dengan nilai inversion rate lainnya. Sedangkan untuk tingkat keberhasilan dalam penemuan solusi masalah, semua nilai inversion rate (antara 0.1 sampai 1) menghasilkan tingkat keberhasilan 100%. Gambar 7 menunjukkan grafik pengujian dengan variasi nilai crossover rate.

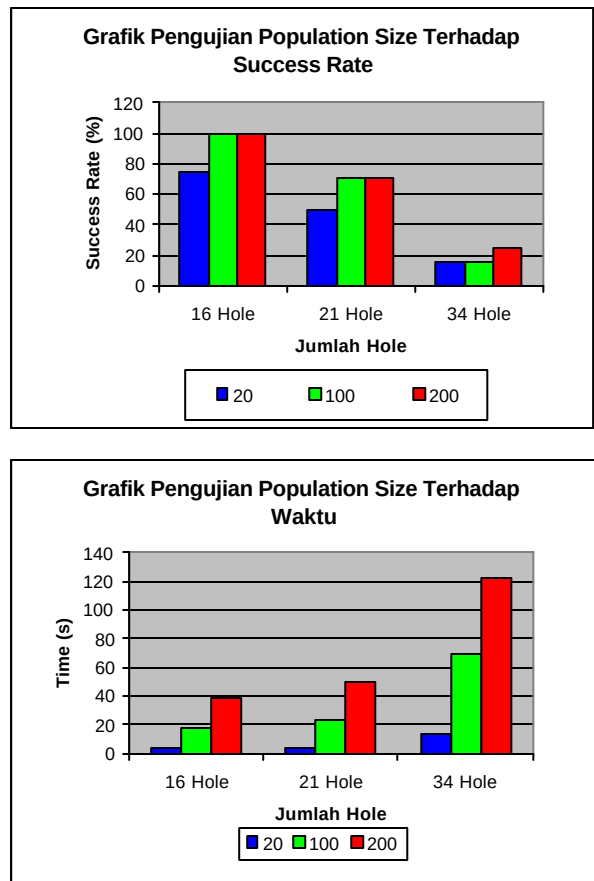


Gambar 7. Hasil Pengujian dengan Variasi Nilai Crossover Rate

Dari hasil pengujian terlihat bahwa untuk nilai crossover rate di bawah 0.5, tingkat keberhasilannya lebih rendah dibandingkan dengan nilai crossover rate di atas 0.5. Dari grafik pengujian juga terlihat bahwa untuk variasi nilai crossover rate dengan nilai inversion rate 0.1 ternyata mencapai tingkat keberhasilan yang lebih rendah dibandingkan dengan nilai inversion rate 0.2 dan 0.3. Namun terlihat juga bahwa semakin tinggi nilai crossover rate maka waktu rata-rata prosesnya akan lebih lama. Sehingga dapat disimpulkan bahwa nilai crossover rate terbaik berkisar antara 0.5 – 0.7, dengan waktu rata-rata prosesnya tidak jauh berbeda dan tingkat keberhasilannya juga tinggi.

2. Pengujian dengan Variasi Population Size

Pengujian ini dilakukan untuk melihat pengaruh jumlah kromosom dalam suatu populasi (*population size*) terhadap proses algoritma genetika. Dalam proses pengujian ini nilai crossover rate yang digunakan adalah 0.6 dan nilai inversion rate yang digunakan adalah 0.2. Gambar berikut adalah grafik pengujian dengan variasi population size.

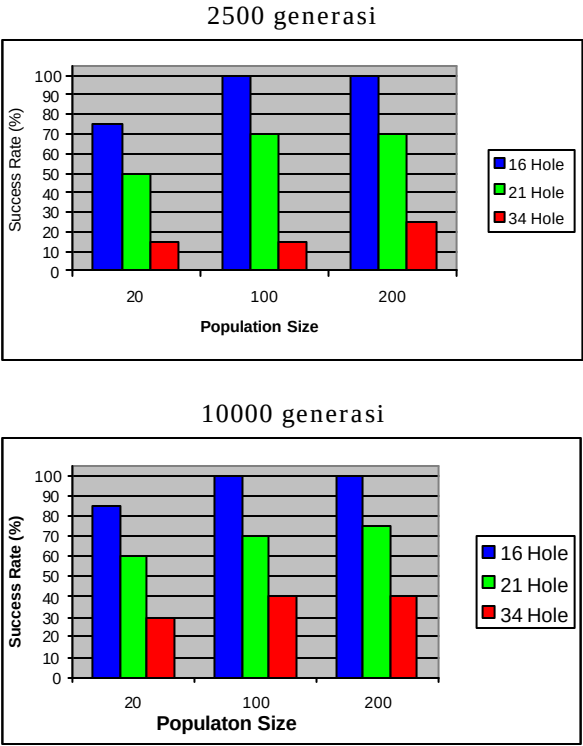


Gambar 8. Grafik Hasil Pengujian dengan Variasi Population Size

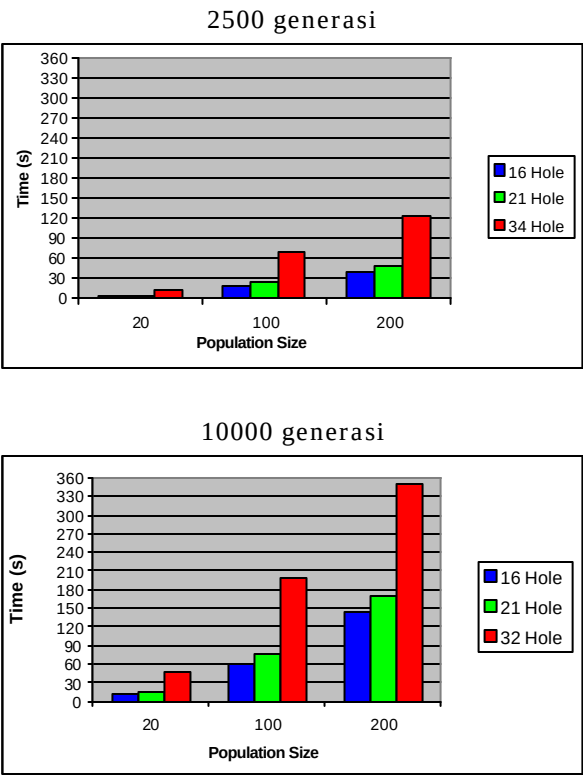
Dari hasil pengujian terlihat bahwa semakin besar nilai population size maka tingkat keberhasilan (*success rate*) algoritma genetika dalam menemukan solusi masalah semakin bagus. Akan tetapi semakin besar nilai population size maka waktu rata-rata yang diperlukan untuk menemukan solusi pun akan semakin lama. Jadi secara keseluruhan dapat diambil kesimpulan bahwa semakin besar *population size* maka tingkat keberhasilan (*success rate*) yang capai akan semakin tinggi, namun waktu yang dibutuhkan akan semakin lama.

3. Pengujian Kriteria Penghentian Generasi

Pengujian ini dilakukan untuk mengetahui bagaimana pengaruh pelebaran generasi pada kriteria berhenti terhadap *success rate* dan waktu rata-rata proses. Berikut adalah data-data hasil pengujian yang ditampilkan dalam bentuk grafik.



Gambar 9. Grafik Hasil Pengujian Kriteria Berhenti untuk 2500 generasi dan 10000 generasi terhadap Success Rate

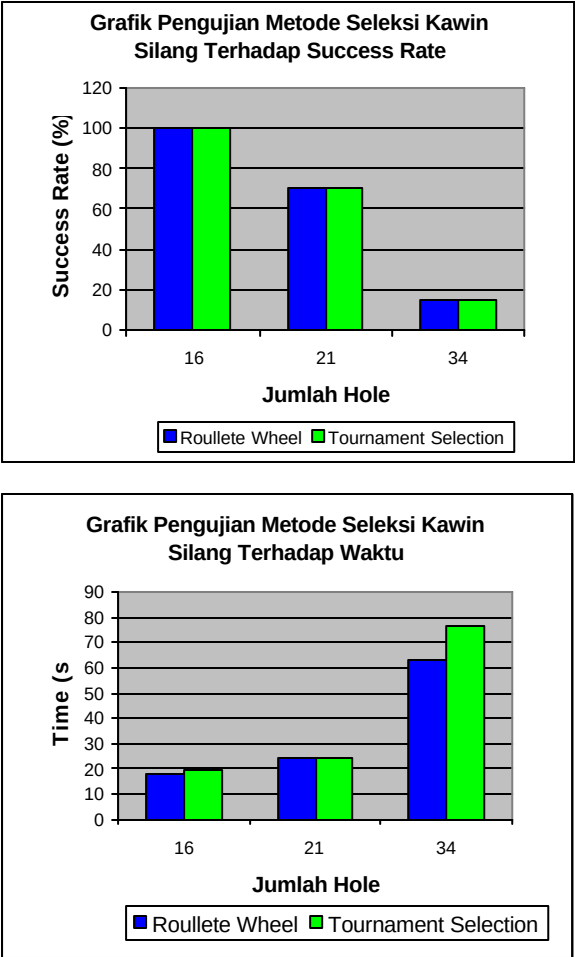


Gambar 10. Grafik Hasil Pengujian Kriteria Berhenti untuk 2500 generasi dan 10000 generasi terhadap Waktu Proses

Dapat disimpulkan dari pengujian bahwa dengan pelebaran jumlah generasi kriteria berhenti maka tingkat keberhasilan yang dicapai akan lebih tinggi namun waktu yang dibutuhkan akan semakin lama.

4. Pengujian Variasi Metode Seleksi

Pengujian ini dilakukan untuk mendapatkan kesimpulan mengenai metode seleksi mana yang bagus. Grafik berikut menunjukkan hasil pengujian yang telah dilakukan.



Gambar 11. Grafik Hasil Pengujian dengan Variasi Metode Seleksi

Dari hasil pengujian ini dapat dilihat bahwa baik metode Roulette Wheel maupun Tournament mempunyai success rate yang sama. Namun waktu rata-rata yang dibutuhkan jika memakai metode Tournament lebih lama sedikit dibandingkan bila memakai metode Roulette Wheel. Jadi dapat dikatakan berdasarkan hasil pengujian bahwa baik metode Roulette Wheel maupun metode Tournament akan memberikan hasil yang tidak jauh berbeda.

5. Pengujian Terhadap Gerakan Mesin Bor Otomatis

Pengujian ini dilakukan untuk membandingkan sistem kerja mesin bor otomatis tanpa penerapan algoritma genetika untuk optimasi gerakan mesin bor otomatis dengan sstem yang menggunakan algoritma genetika. Untuk melihat perbandingan antara pergerakan mesin bor otomatis tanpa algoritma genetika dengan yang menggunakan algoritma genetika dapat dilihat dari efektifitas gerakan antara mesin bor otomatis. Efektifitas gerakan disini maksudnya adalah lama waktu yang dibutuhkan oleh mesin bor otomatis dalam menyelesaikan pengeboran semua pad dan via yang terdapat pada PCB. Hasil pengujiannya dapat dilihat pada Tabel 1.

Tabel 1. Hasil Pengujian Gerakan Mesin Bor Otomatis

PCB	Lama Pengeboran Tanpa Algoritma Genetika (menit)	Lama Pengeboran Dengan Algoritma Genetika (menit)	Selisih (menit)	Persentase (%)
A	16 : 40	07 : 49	08 : 51	53,1
B	13 : 09	09 : 59	04 : 50	36,76
C	19 : 52	12 : 17	07 : 35	38,17

Dari tabel di atas jelas terlihat bahwa waktu yang dibutuhkan untuk melakukan pengeboran dengan menerapkan algoritma genetika lebih cepat dibandingkan dengan pengeboran tanpa algoritma genetika. Perbedaan waktu lama pengeborannya cukup besar. Jadi dapat disimpulkan bahwa algoritma genetika dapat diterapkan untuk mengoptimumkan gerakan mesin bor otomatis.

6. Kesimpulan

Dari hasil pengujian yang telah dilakukan, dapat diambil kesimpulan bahwa algoritma genetika berhasil diterapkan untuk optimasi gerakan mesin bor otomatis dan lebih efisien. Hal ini terbukti lewat pengujian di mana waktu yang dibutuhkan untuk melakukan pengeboran dengan menggunakan algoritma genetika lebih cepat dibandingkan tanpa menggunakan algoritma genetika.

Daftar Pustaka

1. Thiang, Sherwin R.U. Sompie, *Penggunaan Kamera sebagai Sensor Pendeteksi Koordinat Bor pada Mesin Bor Otomatis*, Prosiding Industrial Electronics Seminar, ITS-Surabaya, Oktober 2002.
2. Goldberg, David Edward. *Genetic Algorithms in Search, Optimization, and Machine Learning*. United States of America: Addison-Wesley, 1989.
3. Kendall, Graham. "Artificial Intelligence Methods". *Genetic Algorithms*. United Kingdom. 45pp. 10 Agustus 2002.
4. <<http://www.cs.nott.ac.uk/~gxk/courses/g5baim/004ga/powerpoint/ga.ppt>>
5. Man, K.F, et al. *Genetic Algorithms For Control and Signal Processing*. London: Springer, 1997.
6. Peysakhov, Max. *Genetic Algorithms for the Sub-Graph Isomorphism Problem*. 23 pp. 18 Nov 2002.
7. <http://edge.mcs.drexel.edu/GICL/people/peysakhov/GraphGa/Graph_ga.pdf>
8. Smith, Jeff. *Envolving A Better Solution*. SoftTech Design Inc., 2002. 9 Sept. 2002. <http://www.djonline.com/articles/technology/GeneticAlgorithms_1.asp>.
9. Thiang. *Optimasi Membership Function Fuzzy Logic Controller Dengan Algoritma Genetika*. Prosiding Seminar Intelligent Technology and its Applications (SITIA 2001) ITS-Surabaya, Mei 2001.