

# PSO dengan Dynamic Percepatan Koefisien berdasarkan Multiple Kendala Kepuasan

## Menerapkan Fuzzy Inference System

Chayan Banerjee

Departemen Elektronik & Komunikasi  
Brainware Kelompok Organisasi  
Barasat, India  
chayan.in.2014@ieee.org

Ruchi Sawal

Departemen Elektronika & Instrumentasi  
Karunya Universitas  
Coimbatore, India  
ruchisawal@karunya.edu

**Abstrak - Particle Swarm Optimization (PSO) parameter** seperti Inertia berat badan dan percepatan koefisien umumnya tetap konstan di PSO klasik. Tetapi telah ditemukan bahwa mengubah parameter ini secara dinamis membuat PSO lebih efisien. Dalam makalah ini kami mengusulkan algoritma PSO yang dimodifikasi di mana kita mengubah nilai koefisien percepatan dinamis selama iterasi. Kami telah menggunakan sistem Inference Fuzzy (FIS) untuk mendapatkan nilai koefisien baru untuk PSO untuk setiap putaran. Koefisien tergantung pada kepuasan kendala tertentu yang diberikan sebagai masukan kepada FIS. Modifikasi dinamis ini koefisien telah ditemukan untuk meningkatkan efisiensi PSO dan juga meningkatkan kecepatan konvergensi nya.

**koefisien kata kunci-Percepatan, Kendala Kepuasan Fuzzy Inference System, parameter PSO, Sosial hanya PSO**

### I. saya PENDAHULUAN

Optimasi dapat hanya dinyatakan sebagai matematika memilih yang terbaik di antara satu set alternatif atau solusi terbatas atau tak terbatas. Tujuannya adalah untuk menemukan solusi terbaik  $x$  dari satu set solusi yang mungkin  $X$ , menurut kriteria  $= F$   $\{f_1, f_2, \dots, f_n\}$ . Kriteria ini dinyatakan sebagai fungsi matematika bernama fungsi tujuan [1]. teknik optimasi seperti PSO atau Particle Swarm Optimization milik sekelompok optimasi stokastik. optimasi stokastik (SO) metode adalah mereka yang menghasilkan dan menggunakan variabel acak. Untuk masalah stokastik, variabel-variabel acak muncul dalam rumusan masalah optimasi itu sendiri, yang melibatkan fungsi tujuan acak atau kendala acak [2]. Dalam metode ini setiap solusi potensial disebut partikel dan setiap partikel memiliki nilai fitness yang dihitung dengan fungsi fitness. Nilai fitness adalah salah satu yang dioptimalkan. PSO bekerja dengan menghasilkan nomor acak dari solusi atau partikel. Setiap partikel iteratif diperbarui dengan 2 nilai yang berbeda. Salah satunya adalah nilai terbaik dicapai oleh partikel sejauh (yaitu lokal terbaik atau  $g_{\text{terbaik}}$  value) dan yang lainnya adalah nilai terbaik dicapai oleh partikel apapun sejauh ini (global yang terbaik atau

$g_{\text{terbaik}}$ ). Berdasarkan dua nilai terbaik ini, kecepatan partikel dan posisinya diperbarui dan kebugaran partikel dihitung. Dengan cara ini solusi optimal diperoleh setelah sejumlah iterasi [3].

inferensi fuzzy adalah proses merumuskan pemetaan dari input yang diberikan kepada output menggunakan fuzzy logic. pemetaan kemudian memberikan dasar dari mana keputusan dapat dibuat. Proses inferensi fuzzy melibatkan Keanggotaan Fungsi, Operasi logis, dan Jika-Maka Aturan. Sebuah fungsi keanggotaan (MF) adalah kurva yang mendefinisikan bagaimana setiap titik dalam ruang input dipetakan ke nilai keanggotaan (atau derajat keanggotaan) antara 0 dan 1 [4]. Jika-Maka aturan dan operasi logis (misalnya DAN / ATAU) menentukan output atau inferensi oleh sistem. Karena output umumnya kabur di alam proses defuzzifikasi diikuti untuk mendapatkan output renyah.

Dalam PSO pencarian solusi optimal dipandu oleh 2 komponen percepatan stokastik (lihat gambar 1). Komponen kognitif mengontrol pencarian lokal. Sedangkan komponen sosial mengontrol eksplorasi global partikel. Kontrol yang tepat dari 2 komponen ini sangat penting dalam mencapai solusi optimal secara efisien. [5] Ditegaskannya bahwa nilai yang relatif tinggi dari komponen kognitif akan menghasilkan mengembara berlebihan partikel melalui ruang pencarian sementara nilai yang lebih tinggi dari komponen sosial akan mengarah pada konvergensi prematur partikel [6]. Dalam makalah ini kami menunjukkan bahwa koefisien percepatan berubah secara dinamis membuat PSO lebih efektif dan efisien dalam konvergensi. Di sini kita telah dianggap sebagai pernyataan masalah mengoptimalkan lokasi segerombolan robot sama-abled. Di sini kita menggunakan PSO untuk mengoptimalkan posisi bot tapi dengan menjaga kendala tertentu dalam pikiran. Kami menggunakan sistem inferensi Fuzzy untuk mencatat beberapa tujuan dan memberikan output bernilai tunggal tertentu (mulai 0-4). Kami telah menggunakan crisp output ini dari mesin inferensi sebagai koefisien percepatan untuk algoritma PSO disesuaikan kami.

### II. PERNYATAAN MASALAH

Masalahnya melibatkan sekelompok bots  $B = \{b_1, b_2, \dots, b_n\}$  dengan hardware yang sama dan kemampuan pengolahan. Kepala Cluster  $CH = b_5$  memiliki radius transmisi unit  $r$ . Untuk zaman tertentu kepala klaster mentransmisikan jangkauan transmisi untuk anggota kelompok nya. Sekarang ini bots anggota akan perlu mengoptimalkan posisi mereka sehingga mereka jatuh transmisi

jari-jari  $r$ . Mereka perlu juga diingat 2 kriteria dasar sebelum mereka pindah:

1. Sisa energi dengan bot
2. pemisahan mereka dari radius transmisi saat ini. Kedua kriteria harus

mempengaruhi pergerakan bot.

AKU AKU AKU. C LASSICAL PSO & S Osial HANYA PSO Selama gerakan, partikel memperbarui kecepatan dan posisinya sendiri berdasarkan pengalaman terbaik sendiri dan seluruh penduduk. Langkah-langkah yang terlibat adalah [7]:

#### 1. Kecepatan dan posisi semua partikel yang acak

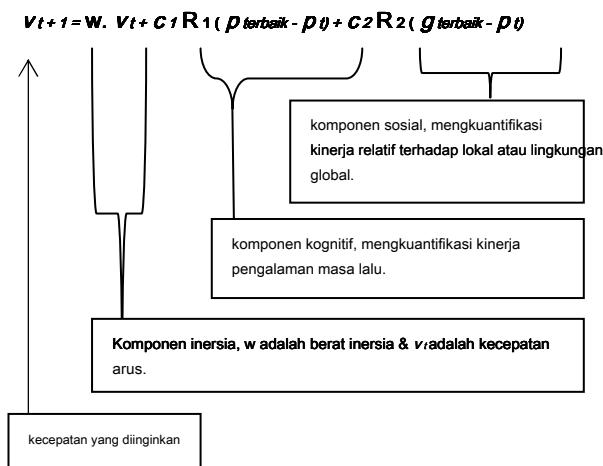
set ke dalam rentang yang telah ditetapkan.

#### 2. Velocity updating- Pada setiap iterasi, kecepatan dari semua

partikel diperbarui menggunakan

$$v_{t+1} = w \cdot v_t + c_1 R_1(p_{\text{terbaik}} - p_t) + c_2 R_2(g_{\text{terbaik}} - p_t) \quad \dots\dots\dots 1$$

**persamaan 1**, di mana  $p_t$  dan  $v_t$  adalah posisi saat ini dan kecepatan partikel partikel. Itu  $p_{\text{terbaik}}$  adalah posisi dengan nilai obyektif terbaik yang ditemukan sejauh ini oleh partikel dan  $g_{\text{terbaik}}$  adalah nilai terbaik ditemukan di seluruh populasi. Jika pikiran dalam pandangan kendaraan atau bot atau partikel mencoba untuk mengoptimalkan posisinya bekerja di segerombolan atau kelompok, maka persamaan di atas dapat dijelaskan sebagai berikut



Gambar. 1, kecepatan PSO persamaan pembaruan.

Memilih parameter PSO [8]: Parameter

adalah  $w$ ,  $c_1$  &  $c_2$

1. ' $w$ ' adalah berat inersia.

$w < 1$ : momentum sedikit yang diawetkan dari langkah waktu sebelumnya.

Perubahan mudah arah mungkin dengan pengaturan ini.  $w = 0$ :

konsep kecepatan benar-benar hilang.

$w > 1$ : sulit untuk mengubah arah karena nilai momentum yang tinggi. Umumnya nilai  $w$  disimpan dalam rentang [0,2, 0,5].

2.  $c_1$  &  $c_2$  keduanya koefisien percepatan Mereka umumnya terletak pada kisaran [0, 4].

#### 3. Posisi updating- posisi semua partikel

diperbarui sesuai dengan.

#### $p_{t+1} = p_t + v_{t+1} \dots\dots\dots$ 2 Update posisi aturan

sederhana, posisi baru adalah penjumlahan dari posisi saat ini dan kecepatan yang diinginkan dihitung 1. Kondisi Berhenti untuk algoritma mungkin jumlah tertentu iterasi yang telah ditetapkan. Banyak varian PSO hybrid telah diajukan untuk mengurangi kompleksitas komputasi yang timbul sementara menerapkan adaptasi dinamis. Tergantung pada penggunaan komponen yang disebutkan dalam gambar 1, PSO yang mungkin memiliki variasi dasar dari bentuk aslinya. PSO mungkin memiliki 3 model kecepatan dasar [8]

##### 1. Kognisi-satunya model:

$$v_{t+1} = w \cdot v_t + c_1 r_1(p_{\text{terbaik}} - x_t)$$

Partikel independen bukit-pendaki Tidak mempertimbangkan keputusan lingkungan.

##### 2. Sosial satunya model:

$$v_{t+1} = w \cdot v_t + c_2 r_2(g_{\text{terbaik}} - x_t)$$

Berikut partikel bukit pendaki stokastik.

##### 3. Model Tanpa Pamrih:

Mengikuti model sosial yang sama dari kasus sebelumnya, tetapi solusi lbest dipilih dari tetangga saja. Partikel itu sendiri diperbolehkan untuk menjadi lbest.

#### IV. D YNAMICALLY berbagai PSO PARAMETER [9] [6] Penawaran

dengan topik variasi parameter di PSO, dimana parameter seperti berat inersia dan percepatan koefisien berubah dengan setiap iterasi. Mereka memiliki manfaat mengeksplorasi ruang pencarian lebih efektif. Koefisien percepatan perubahan dengan iterasi berdasarkan rumus sebagai diberikan di bawah [10]  $c_1 = (c_{1f} - c_{1i}) \text{ iter} / \text{iter}_{\text{max}} + c_{1i} \dots\dots\dots (3) c_2 = (c_{2f} - c_{2i}) \text{ iter} / \text{iter}_{\text{max}} + c_{2i} \dots\dots\dots (4)$  di mana  $c_{1i}$ ,  $c_{1f}$ ,  $c_{2i}$ ,  $c_{2f}$  adalah nilai-nilai awal dan akhir dari faktor percepatan kognitif dan sosial.

[11] [12] Membuat perubahan berat inersia untuk meningkatkan efisiensi dan konvergensi kecepatan algoritma PSO. Dalam makalah ini kami tidak berurusan dengan variasi dinamis dengan berat inersia. Tapi pendekatan berbasis fuzzy yang digunakan di sini dapat juga diterapkan dalam berbagai berat inersia juga. [13] Implements sistem Inference Fuzzy untuk memodifikasi koefisien percepatan. FIS membantu dalam menyesuaikan nilai koefisien percepatan tergantung pada pbest dan nilai-nilai gbest untuk setiap partikel dalam swarm.

#### V. P ROPOSED A pproach

**Asumsi:** Dalam hal pelaksanaan spasial PSO tersebut, pendekatan 2 sumbu dilakukan. optimasi ini dibuat sehubungan dengan lokasi atau koordinat robot. Oleh karena itu harus ada dua set PSO posisi dan kecepatan persamaan, seperti

yang satu persamaan set nilai sumbu x dan mengoptimalkan lainnya nilai sumbu y. Tapi di sini demi kesederhanaan kami telah dianggap hanya satu sumbu untuk optimasi.

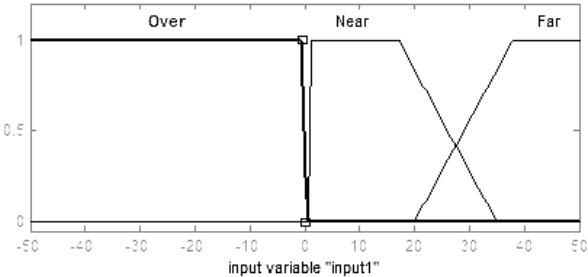
A 2 input, output tunggal Mamdani Jenis Fuzzy Inference System (FIS) digunakan di sini. Output dari FIS dimasukkan ke dalam PSO sebagai koefisien percepatan dan optimalisasi dilakukan. Setelah siklus tertentu atau iterasi adalah atas masukan kepada FIS diperbarui dan Sejalan kita mendapatkan output FIS baru dan koefisien akselerasi juga diperbarui. Berikut dua kendala dianggap dan mereka adalah energi dengan bot untuk jarak perjalanan dan yang lain adalah kedekatan (Jauh, dekat atau di atas) dari bot dengan jari-jari transmisi.

Tabel 1:

| Peraturan No. | Masukan 1- Jarak (Jauh-dekat-Lebih) | Masukan 2- Energi (High-Med-Rendah) | Output-Tendensi (High-Med-Rendah) |
|---------------|-------------------------------------|-------------------------------------|-----------------------------------|
| 1 2           |                                     |                                     |                                   |
| 3 4           |                                     |                                     |                                   |
| 5 6           |                                     |                                     |                                   |
| 7 8           |                                     |                                     |                                   |
| 9             |                                     |                                     |                                   |
|               |                                     |                                     |                                   |
|               |                                     |                                     |                                   |
|               |                                     |                                     |                                   |
|               |                                     |                                     |                                   |

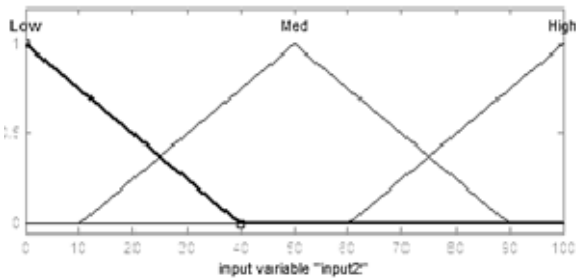
Grafik di atas menggambarkan Aturan Dasar dari sistem inferensi Fuzzy. The 'kecenderungan' seperti yang disebutkan di sini dalam tabel harus diberi makan untuk algoritma PSO yang dimodifikasi dalam bentuk 'koefisien percepatan'. Jadi PSO akan mulai normal dengan nilai yang telah ditetapkan standar. Berikutnya dengan iterasi sebagai partikel tertentu akan bergerak melintasi ruang mengoptimalkan posisinya, koefisien percepatannya juga akan berubah.

PSO yang kami gunakan di sini adalah satu-satunya model Sosial. Dan sehingga tidak ada nilai terbaik lokal 'l' terbaik 'Dalam persamaan pembaruan.

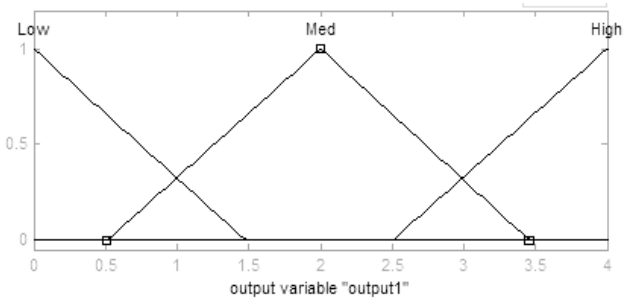


Gambar. 2, fungsi Input1- Keanggotaan

Jadi PSO akhir kecepatan pembaruan aturan adalah sebagai berikut: Velocity  
**Update:**  $v_{i+1} = w \cdot v_{i,j} + CR (g_{i,j} - p_{i,j})$   
Persamaan di atas tidak memiliki komponen kognitif.



Gambar. 3, Input 2- fungsi keanggotaan



Gambar. 4, fungsi output-Keanggotaan

**Komponen sosial adalah sangat penting di sini. g, terbaik adalah nilai terbaik atau posisi (dekat dengan nilai yang diinginkan) dicapai oleh bot dalam kelompok tertentu atau lingkungan. Dalam hal ini g terbaik nilai adalah nilai radius transmisi yang telah dihitung. Rincian yang tidak termasuk dalam makalah ini.**

Jadi kecepatan bot yang diubah dengan aturan pembaruan yang membandingkan lokasi saat ini dengan tujuan an istilah sosial dikalikan dengan koefisien percepatan yang pada dasarnya adalah nilai konstan. Semakin tinggi koefisien tiba-tiba adalah gerakan bot dan lebih adalah percepatan. Sebuah nilai yang lebih rendah dari koefisien akan menyebabkan akselerasi kurang dan gerakan partikel halus. Tapi itu butuh banyak waktu untuk mencapai tujuan dengan akselerasi rendah.

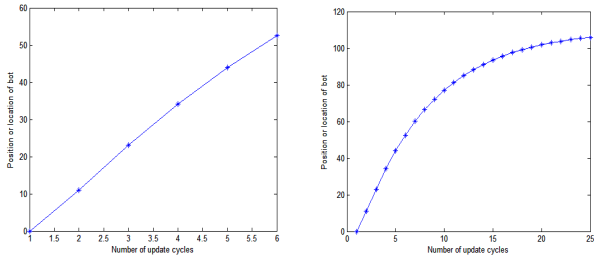
Menggunakan koefisien konstan percepatan atas proses optimasi keseluruhan mungkin tidak menjadi cara yang efisien untuk alasan berikut.

1. nilai yang lebih tinggi dari koefisien percepatan akan memimpin bot untuk overshoot tujuan dan kembali untuk mencapai itu. Ini mengarah ke gerakan yang berlebihan
2. Sebuah nilai yang lebih rendah dari percepatan akan membuat bot lambat dan itu akan memakan waktu lama untuk mencapai tujuan ..
3. Nilai media konservatif akan bagus tetapi akan lagi mencapai tujuan bukan cara terpendek dan akan memiliki kurva ke jalan.
4. Nilai koefisien dinamis yang berubah dengan masing-masing iterasi juga dapat menyimpan dalam pertimbangan beberapa faktor lain yang dapat diambil dari perawatan sebelum memutuskan pindah bot dari satu lokasi ke lokasi lain.

Berikut ini adalah simulasi MATLAB dari PSO yang dimodifikasi, dengan nilai yang berbeda dari koefisien percepatan.

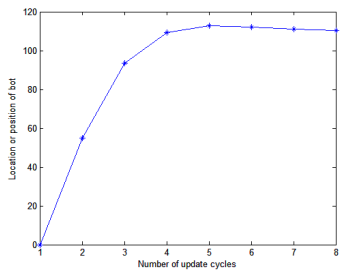
Kasus 1. Dengan  $c = 0,1$ , dan tetap jumlah siklus update (= 6 & 25)

Kedua plot menunjukkan bahwa dengan koefisien rendah akselerasi, lebih banyak waktu-langkah yang dikonsumsi untuk mencapai lokasi tujuan dari 110 unit. Dengan 6 iterasi lokasi tidak tercapai. Dibutuhkan 25 iterasi untuk mencapai lokasi tujuan.

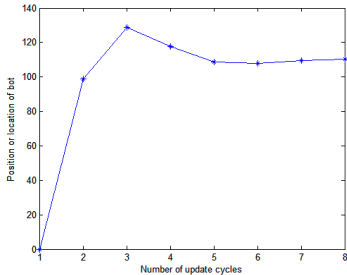


Gambar. 5 PSO dengan siklus pembaruan yang berbeda, 6 dan 20 masing-masing,  $c = 0,1$ , Target Lokasi = 110

Kasus 2. Dengan  $c = 0,5$ , dengan siklus tetap update (= 8) Berbeda dengan jalur yang diambil dengan koefisien akselerasi yang lebih rendah, di sini bot telah melakukan perjalanan rute yang lebih panjang untuk mencapai tujuan.

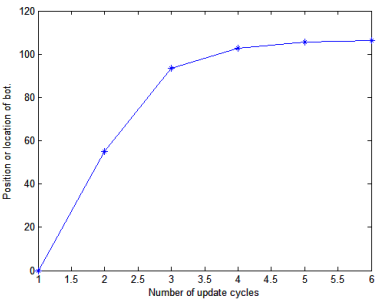


Gambar. 6 Plot PSO dengan  $c = 0,5$ , lokasi target = 110.



Gambar. 7 Plot PSO dengan  $c = 0,5$ , lokasi target = 110.

Kasus 3. Dengan  $c = 0,9$ , dengan jumlah tetap siklus update (= 8).

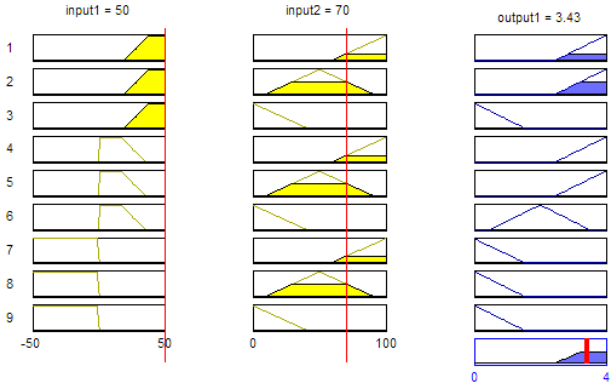


Gambar. 8 Plot PSO menggunakan nilai dinamis  $c$ , lokasi target adalah 110.

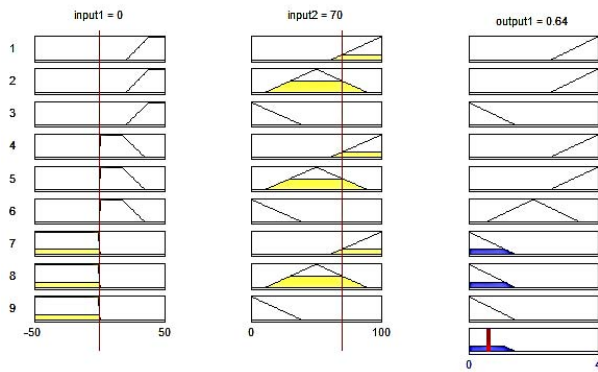
Dengan koefisien akselerasi tinggi bot lampau tujuan dan harus perjalanan jarak ekstra untuk sampai ke tujuan.

Sekarang jika koefisien percepatan dinamis yang digunakan yang akan terus berubah dengan jarak dari bot dari target, menggunakan logika berikut. Peningkatan utama yang dapat diamati dari menggunakan koefisien percepatan variabel adalah pengurangan waktu untuk mencapai tujuan. Pengaturan di atas menempatkan bot di lokasi di 6 langkah waktu, tidak seperti kasus-kasus sebelumnya yang diperlukan 8 sampai 25 langkah waktu atau iterasi untuk mencapai lokasi target.

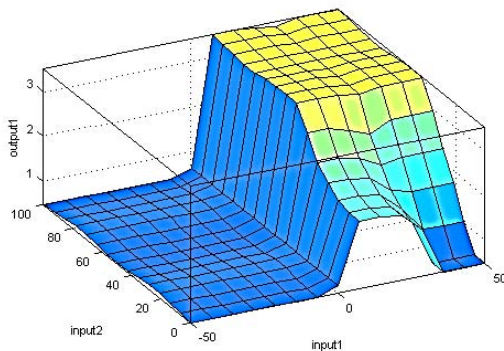
Dasar FIS Aturan dengan dua input, ditunjukkan dengan berbagai masukan, pada Gambar 9 dan 10, tetapi menjaga Energi nilai konstan pada 70% dan hanya input Jarak bervariasi.



Gbr.9 Rule viewer, dengan Far (50), Tinggi (70) sebagai masukan memberikan output tinggi (= 3,43)



Gbr.10 Rule viewer, dengan Over (00), Tinggi (70) sebagai masukan memberikan output tinggi (= 0.64)



Gambar. 11 Permukaan plot dari kemungkinan nilai output terhadap dua input dari FIS.

## VI. C ONCLUSION & F uture W ORK

Sebuah pendekatan baru dari variasi parameter PSO dinamis ditampilkan dalam makalah ini. Sebuah FIS yang mampu mempertimbangkan beberapa masukan / kendala dan memberikan output Sejalan, digunakan. FIS digunakan untuk menghasilkan dinamis koefisien percepatan lebih iterasi dari algoritma PSO.

Hal itu terlihat dari simulasi bahwa koefisien percepatan dinamis yang bervariasi membuat PSO lebih efisien dan selain mengurangi waktu konvergensi.

pekerjaan di masa depan mungkin termasuk pelaksanaan algoritma dimodifikasi dalam hardware dan menganalisa efisiensi yang sama.

## REFERENSI

- [1] Ghanizadeh, A., Sinaie, S., Abarghouei, AA, Syamsuddin, SM, A Fuzzy-partikel swarm optimasi algoritma berdasarkan untuk memecahkan masalah jalan terpendek, 2 Konferensi Internasional tentang Teknik Komputer dan Teknologi, ICCEET 2010, Volume 6, 2010, nomor Pasal 5486167, Pages V6404-V6408
- [2] James C. Spall, Pengantar Stochastic Pencarian dan Optimasi: Estimasi, Simulasi, dan Pengendalian, ISBN: 978-0-471-33052-3 [3] Hossein Yazdani, Halina Kwasnicka, Daniel Ortiz-Arroyo, Proceedings konferensi internasional Ketiga di Computational kolektif intelligen: teknologi dan aplikasi - Volume Bagian I Halaman 224- 233 [4] Fuzzy Proses Inference, pusat Dokumentasi, MathWorks, <http://www.mathworks.in/help/fuzzy/fuzzy-inference-process.html> [5] Kennedy, J. ; Eberhart, R., "Partikel segerombolan optimasi," Neural Jaringan, 1995. Proceedings., IEEE International Conference on, vol.4, tidak ada., Pp.1942,1948 vol.4, November / Desember 1995 [6] Chaturvedi, Krishna Teerth, Manjaree Pandit, dan Laxmi Srivastava. "Partikel segerombolan optimasi dengan waktu yang berbeda-beda koefisien percepatan untuk non-cembung kekuatan ekonomi pengiriman." International Journal of Power & Energi Listrik Sistem 31,6 (2009): 249-257. [7] Pena, Upegui, Sanchez,"Particle Swarm Optimization dengan Discrete Rekombinasi: Sebuah Optimizer Online untuk evolvable Hardware", Prosiding NASA / ESA Konferensi Pertama pada Adaptive Hardware dan Sistem (AHS'06). [8] Swagatam Das, Ajith Abraham, dan Amit Konar, "Particle Swarm Optimasi dan Differential Evolution Algoritma: Technical Analysis, Aplikasi dan Hibridisasi Perspektif", Departemen Elektronika dan Teknik Telekomunikasi, JadavpurUniversity, Kolkata & Centre of Excellence untuk diquantifikasi Kualitas Layanan, Norwegia Universitas Sains dan Teknologi, Norwegia [9]. Tripathi, Praveen Kumar, Sanghamitra Bandyopadhyay, dan Sankar Kumar Pal. "Multi-tujuan optimasi partikel swarm dengan waktu varian inersia dan koefisien percepatan." Ilmu 177,22 (2007): 5033-5049.
- [10] Ratnaweera, Asanga, Saman Halgamuge, dan Harry C. Watson. "Diri-mengorganisir hirarkis partikel swarm optimizer dengan akselerasi waktu yang bervariasi koefisien." Evolutionary Computation, IEEE Transaksi di 8.3 (2004): 240-255. [11] Chatterjee, Amitava, dan Patrick Siarry. "Berat inersia Nonlinear variasi untuk adaptasi dinamis dalam optimasi partikel swarm "Komputer & Riset Operasi 33,3 (2006):... 859-871 [12] Liao, Wudai, Junyan Wang, dan Jiangfeng Wang. "Inersia Nonlinear variasi berat untuk adaptasi dinamis dalam optimasi partikel swarm." Kemajuan dalam Swarm Intelligence. Springer Berlin Heidelberg, 2011. 80-85.
- [13] Liliana, Dewi Yanti, dan M. Rahmat Widyanto. "Particle Swarm Optimasi dengan Akselerasi Adaptive Fuzzy untuk Manusia Obyek Deteksi." International Journal of Video & Image Processing Keamanan Jaringan 11,1 (2011).