

IMPLEMENTASI ALGORITMA PARTICLE SWARM OPTIMIZATION UNTUK OPTIMASI PEMERATAAN GURU MATA PELAJARAN KABUPATEN LUMAJANG

Risda Amalia Khusna¹⁾, Imam Cholissodin, S.Si., M.Kom.²⁾,

Randy Cahya W, S.Kom., M.Kom.³⁾

Program Studi Teknik Informatika

Fakultas Ilmu Komputer

Universitas Brawijaya, Malang 65145, Indonesia

email: risda.a.husna[at]gmail.com¹⁾, imam.cholissodin[at]gmail.com²⁾,

rendicahya[at]ub.ac.id³⁾

ABSTRAK

Persebaran guru yang tidak merata merupakan masalah mendasar yang masih menjadi persoalan mulai tingkat pusat hingga daerah. Persebaran guru yang tidak merata ditandai dengan adanya sekolah yang kekurangan guru dan sekolah kelebihan yang guru. Persolaan tersebut dapat diatasi dengan melakukan penataan dan pemerataan guru berupa mutasi. Mutasi adalah memindah tugaskan guru dari suatu satuan pendidikan ke satuan pendidikan lain. Mutasi harus dilakukan dengan kriteria yang jelas agar penataan dan pemerataan guru dapat dilakukan secara tepat sasaran. Penataan dan pemerataan yang dilakukan secara tepat sasaran akan menjamin guru dapat bekerja lebih baik. Berdasarkan hal tersebut, pada penelitian ini dibuat suatu sistem yang mampu mengoptimasi penataan dan pemerataan guru secara otomatis. Optimasi bertujuan untuk menghasilkan persebaran guru yang lebih proporsional dan lebih tepat sasaran serta sesuai dengan peraturan yang telah ditentukan. Teknik optimasi yang digunakan adalah *Particle Swarm Optimization* (PSO), yaitu teknik optimasi berbasis populasi yang terinspirasi oleh perilaku sosial gerakan kawanan hewan seperti burung atau ikan. PSO menggabungkan metode pencarian lokal dan metode pencarian global, masing-masing partikel akan bergerak mengitari ruang pencarian dan menyesuaikan posisinya terhadap pengalaman pribadi maupun pengalaman partikel lain di sekitarnya. Dari hasil pengujian didapatkan parameter-parameter PSO yang paling optimal untuk iterasi maksimum=70, ukuran populasi=25, $\theta_{max} = 0.9$, $\theta_{min} = 0.4$, $c_{1i} = 2.5$, $c_{1f} = 0.5$, $c_{2i} = 0.5$, dan $c_{2f} = 2.5$. Optimasi dengan menggunakan nilai parameter optimal PSO dapat meningkatkan nilai *fitness* sebesar 8-12%.

Kata kunci: Penataan dan Pemerataan Guru, Kelebihan dan Kekurangan Guru, Optimasi, Algoritma *Particle Swarm Optimization*.

ABSTRACT

Maldistribution of teachers is a fundamental problem that is still become an issue from central to the regional level. This maldistribution is characterized by a teacher shortages in some schools, but overages in other. These issues can be overcome by making the arrangement and equal distribution of teachers, called as displacement. Displacement are redeployed teachers from school to another school. This removal must be done with clear criteria, so that the arrangement and equal distribution of teachers can be done right on target. Structuring and equalization which done on the right target will ensure teachers can work better. According to these, this study will created a system that is able to provide structuring and optimization of equal distribution of teachers automatically. Optimization aims to produce a more proportional distribution of teachers and better right on target and in accordance with predetermined rules. The optimization technique used is Particle Swarm Optimization (PSO), it is a population-based optimization techniques inspired by the social behavior of animals flock movements like a bird or a fish. PSO combines methods of local search and global search methods, each particle will move around the search space and adjust the position of the personal experiences nor experiences of other particles in the surrounding. The test results obtained the most optimal parameters of PSO, that is maximum iteration = 70, population size=25, $\theta_{max} = 0.9$, $\theta_{min} = 0.4$, $c_{1i} = 2.5$, $c_{1f} = 0.5$, $c_{2i} = 0.5$, and $c_{2f} = 2.5$. Optimization using PSO optimal parameter can improve the fitness by 8-12%.

Keywords: Arrangement and Equal Distribution of Teachers, Teacher Shortages and Overages, Optimization, Particle Swarm Optimization Algorithm.

1. PENDAHULUAN

Pemerataan guru merupakan salah satu upaya untuk meningkatkan mutu pendidikan yang dilakukan dengan cara mutasi guru. Mutasi adalah pemindahan guru antar satuan pendidikan dalam lingkup kecamatan, kabupaten, atau provinsi agar komposisi guru sesuai dengan kebutuhan riil setiap satuan pendidikan. Namun, pada kenyataannya upaya pemerataan guru dengan cara mutasi belum bisa memberikan hasil sesuai dengan harapan. Sampai saat ini masih dapat ditemui persebaran guru yang belum merata pada beberapa daerah di Indonesia. Banyak sekolah yang kekurangan guru, tetapi disisi lain juga ada sekolah yang kelebihan guru. Menteri Pendidikan dan Kebudayaan Muhammad Nuh menegaskan bahwa kekurangan guru terjadi bukan karena tidak tersedianya guru yang mencukupi, melainkan lebih kepada persebaran guru yang belum merata. Mengangkat guru baru hanya akan memberikan solusi untuk sekolah yang kekurangan guru namun tidak untuk sekolah yang kelebihan guru, selain itu pengangkatan guru baru akan membutuhkan anggaran dana yang lebih banyak. Sehingga, lebih baik dilakukan optimalisasi jumlah guru yang ada untuk didistribusikan merata secara nasional (Akuntono, 2011).

Berdasarkan adanya kebutuhan untuk optimalisasi pemerataan guru maka diperlukan suatu sistem cerdas untuk melakukan mutasi guru secara otomatis dengan menghasilkan persebaran guru yang lebih seimbang dan sesuai dengan peraturan yang telah ditentukan sehingga dapat meringankan proses mutasi yang selama ini masih dilakukan secara manual. Salah satu teknik yang sering digunakan dalam masalah optimasi adalah Particle Swarm Optimization (PSO). PSO adalah teknik optimasi berdasarkan populasi yang terinspirasi oleh perilaku sosial gerakan kawanan hewan seperti burung atau ikan dimana setiap objek hewan seperti dianggap sebagai sebuah partikel. PSO menggabungkan metode pencarian lokal dan metode pencarian global karena posisi setiap partikel akan berubah terhadap waktu dimana masing-masing partikel akan bergerak mengitari ruang pencarian dan menyesuaikan posisinya terhadap pengalaman pribadi maupun pengalaman partikel lain di sekitarnya (Santosa, 2011).

Algoritma PSO telah banyak digunakan dalam penelitian yang berkaitan dengan optimasi penentuan penempatan suatu objek untuk mencapai tujuan atau keuntungan tertentu. PSO sendiri telah terbukti mampu mengatasi berbagai permasalahan kombinatorial kompleks dan mempunyai area

pencarian yang luas (Mahmudy, 2015). Maka pada penelitian ini, peneliti menggunakan algoritma PSO untuk mengoptimasi pemerataan guru mata pelajaran di Kabupaten Lumajang agar persebaran guru di Kabupaten lumajang lebih merata dan lebih tepat sasaran.

2. PERMASALAHAN

Dari paparan pendahuluan, penelitian ini merumuskan permasalahan bagaimana mengimplementasikan algoritma *Particle Swarm Optimization* untuk mengoptimasi pemerataan guru mata pelajaran di Kabupaten Lumajang dan bagaimana hasil pengujian dari implementasinya

3. DASAR TEORI

Guru

Guru adalah pendidik profesional dengan tugas utama mendidik, mengajar, membimbing, mengarahkan, melatih, menilai, dan mengevaluasi peserta didik pada pendidikan anak usia dini jalur pendidikan formal, pendidikan dasar, dan pendidikan menengah. Berdasarkan sifat, tugas, dan kegiatannya, guru dibedakan menjadi 3 jenis yakni sebagai berikut (Kemenristekdikti, 2011) :

1. Guru Kelas

Guru Kelas adalah guru yang mempunyai tugas, tanggung jawab, wewenang, dan hak secara penuh dalam proses pembelajaran seluruh mata pelajaran di kelas tertentu pada satuan pendidikan formal tingkat (Taman Kanak-kanak)TK/ (Taman Kanak-Kanak Luar biasa) TKLB dan SD/(Sekolah Dasar Luar Biasa) SDLB atau satuan pendidikan formal lain yang sederajat.

2. Guru Mata Pelajaran

Guru Mata Pelajaran adalah guru yang mempunyai tugas, tanggung jawab, wewenang, dan hak secara penuh dalam proses pembelajaran mata pelajaran tertentu pada satuan pendidikan formal tingkat SMP/(Sekolah Menengah Pertama Luar biasa) SMPLB, (Sekolah Menengah Atas) SMA/ (Sekolah Menengah Atas Luar Biasa)SMALB, dan (Sekolah Menengah Kejuruan) SMK. Termasuk guru mata pelajaran pendidikan jasmani dan kesehatan, dan guru pendidikan agama pada satuan pendidikan formal tingkat SD/SDLB.

3. Guru Bimbingan dan Konseling

Guru Bimbingan dan Konseling adalah guru yang mempunyai tugas, tanggung jawab, wewenang, dan hak secara penuh dalam kegiatan bimbingan dan konseling terhadap beberapa peserta didik pada satuan pendidikan formal tingkat SMP/ SMPLB ,SMA/SMALB dan SMK.

Mutasi

Mutasi guru adalah perpindahan guru antar satuan pendidikan dalam satu kabupaten/kota, dalam kabupaten/kota yang berbeda, maupun dalam provinsi yang berbeda dikarenakan sebab tertentu. Secara umum penyebab terjadinya mutasi adalah (Kemenristekdikti, 2011):

1. Mutasi karena kenaikan jabatan atau disebut dengan mutasi promosi
2. Mutasi karena adanya kebijakan bupati untuk pergantian formasi dalam rangka melakukan pemerataan guru.
3. Mutasi karena pindah tempat tinggal.

Mutasi Guru PNS

Pada prinsipnya mekanisme mutasi guru PNS antar satuan pendidikan pada kabupaten/kota yang sama, pada kabupaten/kota lain dalam provinsi yang sama, maupun pada kabupaten/kota dalam provinsi yang berbeda mengacu pada ketentuan dan peraturan yang berlaku. Agar tujuan pemerataan guru PNS dapat tercapai maka pemerintah setempat wajib memfasilitasi proses pemindahan guru (kemenristekdikti, 2011). MPLB, SMA/SMALB dan SMK.

Kriteria Guru PNS dalam Mutasi

Ada beberapa kriteria yang menjadikan guru yang berstatus PNS untuk dimutasi, diantaranya adalah (Kemenristekdikti, 2011) :

1. Guru yang bertugas di satuan pendidikan dengan kelebihan guru
2. Guru yang dibutuhkan oleh satuan pendidikan lain karena mempunyai keterampilan atau keahlian khusus
3. Guru yang mempunyai sertifikat pendidik tetapi belum da memenuhi beban mengajar minimal 24 jam per minggu
4. Lebih diprioritaskan untuk guru yang masa kerjanya paling sedikit
5. Permintaan mutasi dari guru sendiri
6. Lebih diprioritaskan untuk guru yang memiliki jarak terdekat dengan satuan pendidikan yang kekurangan guru

Waktu Pemindahan

Demi menjaga keberlangsungan proses pembelajaran di sekolah, guru yang dimutasi harus sudah mengajar di sekolah yang baru mulai semester awal. Jika mutasi guru dilakukan dalam kabupaten/kota yang sama maka mutasi dilakukan di akhir semester pada tahun berjalan dan jika mutasi guru dilakukan dalam kabupaten/kota yang berbeda, maka mutasi guru harus dilakukan paling

lambat pada bulan Oktober tahun berjalan (Kemenristekdikti, 2011).

Optimalisasi Pemenuhan Kebutuhan Guru Mata Pelajaran

Optimalisasi pemenuhan kebutuhan guru mata pelajaran adalah usaha untuk meminimalkan adanya kekurangan dan kelebihan guru mata pelajaran dengan cara memindahkan guru mata pelajaran tertentu dari satuan pendidikan ke satuan pendidikan lain yang kekurangan guru dalam kabupaten/kota yang sama atau dalam kabupaten/kota yang berbeda. Jika memungkinkan untuk memindahkan guru ke sekolah dengan kekurangan mata pelajaran yang bukan bidangnya, maka pemindahan guru dapat dilakukan dengan mempertimbangkan kedekatan latar belakang pendidikan dengan mata pelajaran yang akan diampu atau bisa disebut alih fungsi/profesi. Guru alih fungsi/profesi akan diikutsertakan dalam pendidikan/pelatihan/penataran agar mendapatkan kompetensi profesional pada mata pelajaran baru yang akan diampu (Kemenristekdikti, 2011).

Particle Swarm Optimization (PSO)

Algoritma PSO adalah algoritma yang terinspirasi oleh perilaku sosial sekawanan hewan seperti burung atau ikan. Seekor hewan dalam PSO dianggap sebagai partikel. Perilaku partikel dipengaruhi oleh kecerdasan dari partikel itu sendiri dan perilaku dari partikel lain di kelompoknya. Jika satu partikel menemukan jalan tepat dan pendek menuju ke suatu sumber makanan, maka partikel partikel yang lain akan mengikuti jalan tersebut. Pada PSO, kawanan (swarm) mempunyai ukuran tertentu dimana posisi awal setiap partikel terletak di suatu lokasi yang acak pada suatu ruang multidimensi. Setiap partikel mempunyai 2 karakteristik yaitu posisi dan kecepatan. Setiap partikel bergerak dalam ruang tertentu dengan cara mengingat posisi terbaik yang pernah ditemui dan menyampaikan informasi mengenai posisi terbaik tersebut kepada partikel lain agar semua partikel dapat menyamakan kecepatan antara satu dengan yang lainnya. Hal ini bisa digambarkan seperti pada perilaku burung saat bersama kawanannya, beberapa perilaku tersebut antara lain (Santosa, 2011):

1. Jarak antar burung tidak terlalu dekat (kohesi)
2. Arah terbang burung sesuai dengan arah rata-rata keseluruhan burung yang ada pada kelompoknya (*alignment*)
3. Posisi burung akan menyesuaikan posisi rata-rata burung yang lain dengan (separasi)

Pencarian solusi dalam algoritma PSO dilakukan oleh *swarm*. Inisialisasi *swarm* dibangkitkan secara acak dengan batasan nilai

terkecil dan terbesar dari dimensinya. Partikel-partikel dalam populasi merepresentasikan solusi untuk permasalahan yang dihadapi. Pencarian solusi paling optimal dilakukan partikel dengan melintasi ruang pencarian. Saat melintasi ruang pencarian, partikel akan menyesuaikan posisi terbaik dirinya sendiri (P_{best}) dan posisi terbaik seluruh partikel di kelompoknya (G_{best}) hingga mencapai batas maksimum iterasi atau didapat posisi yang konvergen. Solusi yang dipresentasikan oleh partikel di setiap iterasi akan dievaluasi menggunakan fungsi objektif yang telah ditetapkan. Posisi partikel akan semakin mengarah ke target yang dituju (minimasi atau maksimasi fungsi) di setiap iterasinya. Pada PSO, P_{best} dan G_{best} akan disimpan nilainya untuk keseluruhan iterasi. Secara singkat, tahapan algoritma PSO terdiri dari (Santosa, 2011):

1. Membangkitkan posisi awal sejumlah partikel lengkap dengan kecepatan awal secara acak
2. Mengevaluasi *fitness/cost* posisi setiap partikel
3. Menentukan P_{best} dan G_{best}
4. Memperbarui kecepatan setiap partikel iterasi ke- $t+1$ menggunakan Persamaan 1

$$v_{ij}(t) = v_{ij}(t-1) + c_1 r_1 \quad (1)$$

$$[x_{ij}^L - x_{ij}(t-1)] + c_2 r_2$$

$$[x^G - x_{ij}(t-1)]$$

$$x_{ij}(t) = x_{i1}(t), x_{i2}(t), \dots, x_{ij}(t)$$

$$v_{ij}(t) = v_{i1}(t), v_{i2}(t), \dots, v_{ij}(t)$$

Keterangan:

$x_{ij}(t)$ = posisi partikel ke- i , dimensi ke- j pada iterasi ke- t

$v_{ij}(t)$ = kecepatan partikel ke- i , dimensi ke- j pada iterasi ke- t

$x_{ij}^L = P_{best}$ dari partikel ke- i

$x^G = G_{best}$ dari seluruh swarm

c_1 = konstanta *learning factor* untuk partikel

c_2 = konstanta *learning factor* untuk swarm

r_1, r_2 = bilangan acak pada antara 0 - 1

5. Memperbarui Posisi setiap partikel iterasi ke- $t+1$ menggunakan Persamaan 2.
6. Mengevaluasi *fitness/cost* posisi setiap partikel pada iterasi ke- $t+1$
7. Menentukan P_{best} dan G_{best} di iterasi ke- $t+1$
8. Mengulangi langkah 4 sampai langkah 7 hingga kondisi berhenti terpenuhi.

$$x_{ij}(t) = x_{ij}(t-1) + v_{ij}(t) \quad (2)$$

Inisialisasi Partikel

Posisi awal sebuah partikel harus berada dalam *range* $[x_{min}, x_{max}]$. Jika posisi yang dihasilkan kurang dari x_{min} maka posisi di kembalikan kepada x_{min} , dan jika posisi melebihi x_{max} maka posisi dikembalikan pada x_{max} . Formula untuk inisialisasi partikel sesuai dengan *range* dimensinya dapat dituliskan menggunakan Persamaan 3.

$$x_{ij}(t) = x_{min,j} + r(x_{max,j} - x_{min,j}) \quad (3)$$

Keterangan :

x_{max} = posisi maksimum partikel pada dimensi ke- j

x_{min} = posisi minimum partikel pada dimensi ke- j

r = angka random 0-1

Banyaknya partikel sangat berpengaruh terhadap hasil optimasi yang dilakukan. Pada kebanyakan masalah, penggunaan 10 partikel sudah cukup untuk mendapatkan hasil yang bagus. Namun, untuk hasil yang lebih bagus bisa digunakan 20 sampai 40 partikel. Semakin kompleks masalah yang dihadapi bisa digunakan partikel yang lebih banyak lagi.

Bobot Inersia

Untuk mengatasi update kecepatan yang terlalu cepat pada algoritma PSO dilakukan modifikasi atau perbaikan dengan menambahkan suatu variabel pada formula *update* kecepatan yang disebut dengan θ (bobot inersia). *Update* kecepatan yang terlalu cepat akan menyebabkan nilai optimal dari fungsi tujuan yang dicari sering terlewatkan. Sehingga, perbaikan formula *update* kecepatan pada PSO dapat dituliskan seperti Persamaan 4.

Diperlukan untuk menentukan nilai θ yang mampu menjaga keseimbangan antara pencarian global dan lokal dikarenakan nilai θ yang terlalu tinggi akan menambah porsi pencarian global dan untuk nilai θ yang rendah akan lebih menekankan pencarian lokal. Oleh karena itu, ditentukan suatu Persamaan seperti yang dituliskan pada Persamaan 5 untuk menentukan nilai θ agar nilainya semakin mengecil seiring dengan bertambahnya iterasi dan dapat digunakan mempercepat konvergensi (Santoso, 2011). Persamaan 5 bisa disebut juga dengan Persamaan *Time Varying Inertia Weight* (TVIW).

$$v_{ij}(t) = \theta v_{ij}(t-1) + c_1 r_1 \quad (4)$$

$$[x_{ij}^t - x_{ij}(t-1)] + c_2 r_2$$

$$[x^G - x_{ij}(t-1)]$$

$$\theta_t = \theta_{max} - \left(\frac{\theta_{max} - \theta_{min}}{t_{max}} \right) t$$

atau

$$\theta_t = \theta_{min} + (\theta_{max} - \theta_{min}) * \left(\frac{t_{max} - t}{t_{max}} \right) \quad (5)$$

Keterangan :

θ_{min} = nilai awal

θ_{max} = nilai akhir

biasanya digunakan $\theta_{min} = 0.4$ dan $\theta_{max} = 0.9$

t = iterasi sekarang

t_{max} = iterasi maksimum

Penentuan nilai θ dapat dilakukan dengan beberapa cara diantaranya adalah sebagai berikut (Huang, 2008):

1. Konstan

θ akan ditentukan menggunakan nilai antara 0-1

2. Penurunan

Nilai awal θ dikurangi nilai akhir θ secara linear menggunakan Persamaan 2.8, biasanya digunakan nilai awal $\theta = 0.9$ dan nilai akhir $\theta = 0.4$

3. Acak

θ ditentukan menggunakan Persamaan 6

$$\theta = 0.5 - \left(\frac{rnd}{2.0} \right) \quad (6)$$

Keterangan :

rnd = nilai acak antara 0-1.

Pembatasan Kecepatan

Setiap nilai kecepatan yang didapatkan harus dipastikan berada dalam *range* $[-v_{max}, v_{max}]$, dimana v_{max} merupakan kecepatan maksimum yang diperbolehkan pada suatu dimensi. *Range* $[-v_{max}, v_{max}]$ bertujuan untuk mencegah partikel meninggalkan daerah pencarian. Jika kecepatan yang dihasilkan kurang dari $-v_{max}$ maka nilai kecepatan di kembalikan kepada $-v_{max}$, dan jika kecepatan melebihi v_{max} maka nilai kecepatan dikembalikan pada v_{max} . Pada umumnya, besar nilai v_{max} adalah 10-20% dari posisi terbesar yang bisa diambil pada suatu dimensi (Eberhart & Shi,

2001). Perhitungan v_{max} bisa dilakukan dengan menggunakan Persamaan 7.

$$v_{max} = (n\% * x_{max}) \quad (7)$$

Keterangan :

$n\%$ = banyaknya prosentase yang digunakan

Kondisi Berhenti

Kondisi berhenti dalam PSO merupakan syarat yang digunakan untuk mengakhiri iterasi pencarian. Beberapa syarat pemberhentian iterasi yang bisa digunakan dalam PSO adalah sebagai berikut :

4. Iterasi berhenti apabila iterasi telah mencapai maksimum iterasi
5. Iterasi berhenti apabila ditemukan solusi yang telah memenuhi kriteria
6. Iterasi berhenti apabila tidak ada perubahan nilai atau disebut dengan kondisi konvergen
7. Iterasi berhenti apabila nilai radius swarm yang dinormalisasi mendekati 0
8. Iterasi berhenti apabila grafik fungsi obyekif mendekati 0 seiring bertambahnya iterasi.

Time Varying Acceleration Coefficients (TVAC)

TVAC berguna untuk menyeimbangkan ruang pencarian antara *global exploration* dengan *local exploitation* dengan cara menentukan nilai c_1 menggunakan Persamaan 8 dan menentukan nilai c_2 menggunakan Persamaan 9. Dalam TVAC, nilai c_{1i} ke c_{1f} akan semakin menurun nilai dari c_{2i} ke c_{2f} akan semakin meningkat seiring bertambahnya iterasi. Dengan kondisi nilai komponen cognitive lebih besar dari komponen sosial di awal iterasi mengujikan partikel-partikel untuk bergerak mengelilingi ruang pencarian daripada bergerak menuju nilai populasi terbaik sebaliknya dengan nilai komponen cognitive lebih kecil dari komponen sosial pada iterasi akhir maka akan membuat partikel-partikel bertemu pada nilai global optimum pada proses terakhir optimasi.

$$c_1 = (c_{1f} - c_{1i}) * \frac{t}{t_{max}} + c_{1i} \quad (8)$$

$$c_2 = (c_{2f} - c_{2i}) * \frac{t}{t_{max}} + c_{2i} \quad (9)$$

4. METODE PENELITIAN

Pada penelitian dalam bidang perangkat lunak, umumnya menggunakan tahapan-tahapan

penelitian. Gambar 1 adalah tahapan-tahapan penelitian untuk membangun sistem *clustering* jurnal internasional yang dapat digunakan sebagai alat rekomendasi publikasi berdasarkan laboratorium peneliti.



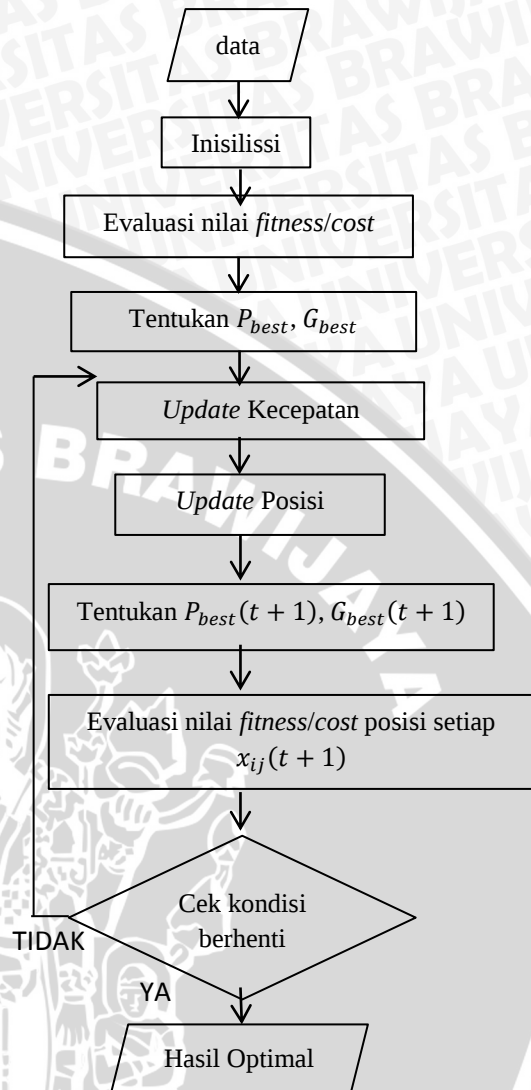
Gambar 1. Tahapan-tahapan penelitian

Berdasarkan gambar 1, tahapan-tahapan yang dilakukan dalam penelitian ini adalah :

1. Mencari dan mempelajari teori-teori yang berkaitan dengan topik penelitian penulis. Teori-teori tersebut dapat berasal dari jurnal, paper, buku, dan e-book.
2. Menentukan kebutuhan untuk membangun sistem. Kebutuhan untuk membangun sistem optimasi pemerataan mutasi guru menggunakan algoritma PSO.
3. Mengumpulkan data yang dibutuhkan untuk implementasi sistem.
4. Merancang sistem yang akan dibangun. Perancangan dilakukan setelah semua kebutuhan sudah ditetapkan pada saat tahap analisis kebutuhan.
5. Mengimplementasi sistem dengan mengacu pada perancangan sistem.
6. Melakukan pengujian dengan tujuan untuk mengetahui apakah algoritma PSO yang diimplementasikan dalam program sudah benar dan bertujuan untuk mengetahui nilai terbaik dari parameter-parameter PSO.
7. Memberikan kesimpulan dan saran. Kesimpulan bertujuan untuk menjelaskan secara singkat sistem yang dibangun dan hasil pengujian sistem yang diperoleh. Saran ditujukan untuk pengembangan penelitian kedepannya agar menjadi lebih baik.

Berdasarkan tahapan penelitian yang disebutkan, maka dibangun sebuah sistem optimasi pemerataan guru mata pelajaran menggunakan

algoritma PSO dengan diagram alir sistem ditunjukkan pada Gambar 2.



Gambar 2. Diagram Alir Algoritma PSO

Berdasarkan diagram alir pada Gambar 2, algoritma PSO terdiri dari proses yang meliputi :

1. Inisialisasi partikel

Inisialisasi partikel digunakan untuk menentukan posisi awal setiap partikel pada suatu populasi (swarm) yang dilakukan secara acak menggunakan persamaan 4. Pada tahap inisialisasi partikel, kecepatan awal untuk setiap partikel bernilai 0.

Contoh Perhitungan:

Misalkan suatu mata pelajaran 8 pilihan dengan $x_{min} = 1$, $x_{max} = 8$ (disesuaikan dengan banyak pilihan), dan nilai r dimisalkan adalah 0.4, maka posisi yang didapatkan adalah sebagai berikut :

$$\begin{aligned}
 x_i &= 1 + 0.4 * (8 - 1) \\
 &= 1 + 0.4 * 7
 \end{aligned}$$

$$= 3.8 \approx 4$$

2. Evaluasi nilai *fitness*

Setelah proses inialisasi partikel selesai langkah selanjutnya adalah menghitung nilai *fitness* dari masing-masing partikel dengan menggunakan Persamaan 10.

$$Fitness = mk + \frac{1}{u+j} \quad (10)$$

Keterangan :

mk = masa kerja

u = usia

j = jarak

3. Penentuan P_{best} dan G_{best}

P_{best} adalah posisi terbaik yang pernah dicapai oleh partikel pada iterasi sebelumnya dan G_{best} adalah posisi terbaik untuk semua partikel yang ditemukan sampai iterasi ke- i . Pada kasus ini, G_{best} menunjukan populasi dengan nilai total *fitness* terbesar yang ditemukan sampai itersai yang telah ditentukan. P_{best} setiap partikel pada iterasi ke 0 sama dengan posisi awal karena iterasi 0 merupakan iterasi pertama

4. Update Kecepatan

Update kecepatan dilakukan dengan menggunakan Persamaan 3 .
Contoh Perhitungan:

Misalkan akan mengupdate kecepatan partikel pada iterasi 1 dengan bobot inersia =0.7333 kcepatan=0, $c_1=1.8333$ $c_2=0.7222$, $r_1=r_2= 0.5$, range kecepatan $[-4.8 - 4.8]$, range posisi $[1,8]$, mksimum iterasi= 3.Maka nilai update kecepatan diperoleh dengan perhitungan sebagai berikut :

$$\begin{aligned} v_{1,1}(1) &= (0.7333 * 0) + (1.8333 * 0.5) \\ &= (4 - 4) + (0.7222 * 0.5)(2 - 4) \\ &= -0.75833 \end{aligned}$$

Nilai bobot inersia yang digunakan dalam mengitung *update* kecepatan diperoleh dari Persamaan 5. maka perhitungannya adalah sebagai berikut

$$\begin{aligned} \theta_1 &= 0.4 + (0.9 - 0.4) * \left(\frac{3-1}{3}\right) \\ &= 0.7333 \end{aligned}$$

Nilai, c_1 yang digunakan dalam mengitung *update* kecepatan diperoleh dari Persamaan 8, sedangkan nilai c_2 dihitung menggunakan Persamaan 9. dengan nilai $c_{1i} = 2.5$, $c_{1f} =$

0.5 , $c_{2i} = 0.5$, $c_{2f} = 2.5$. Maka perhitungannya adalah sebagai berikut :

$$c_1 = (0.5 - 2.5) * \left(\frac{1}{3}\right) + 2.5 = 1.8333$$

$$c_2 = (2.5 - 0.5) * \left(\frac{1}{3}\right) + 0.5 = 0.7222$$

Penentuan *range* kecepatan yang digunakan dalam mengitung update kecepatan diperoleh dari Persamaan 7 . maka perhitungannya adalah sebagai berikut:

$$v_{max} = \left(\frac{60}{100} * 8\right) = 4.8$$

Nilai 8 merupakan x_{max} atau posisi terbesar yang bisa digunakan jadi kecepatan sebesar -0.75833 masih dalam *range* $[-4.8; 4.8]$.

5. Update Posisi

Udate posisi dilakukan untuk mendapatkan posisi baru dengan menggunakan Persamaan 2 .
Contoh perhitungan:

Misalkan nilai update kecepatan yang diperoleh sebesar dengan nilai $x_{min} = 1$ dan $x_{max} = 8$, maka posisi baru didapatkan menggunakan perhitungan sebagai berikut :

$$x_{1,1} = 4 + -0.75833 = 3.041666667 \approx 3$$

Kemudian dilakukan pengecekan terhadap posisi yang baru. Posisi yang didapatkan yaitu 3 masih dalam range posisi antra x_{min} dan x_{max} karena tidak kurang dari 1 dan tidak lebih dari 8.

6. Penentuan P_{best} (t+1) dan G_{best} (t+1)

Penentuan P_{best} dan G_{best} iterasi selanjutnya, dimisalkan akan mencari pbest dan gbest sampai pada iterasi 1 maka caranya adalah dengan mencari populasi dengan nilai *fitness* terbaik pada dari iterasi 0 dan iterasi 1 sedangkan G_{best} merupakan populasi yang memiliki nilai *fitness* terbesar dari P_{best} yang diadptkan sebelumnya

7. Pemberhentian iterasi

Dimisalkan syarat pemberhentian berdasarkan maksimum iterasi, jika iterasi dilakukan sebanyak 3 kali , maka pencarian solusi kan berhenti setelah mencapai iterasi ke 3

8. HASIL DAN PEMBAHASAN

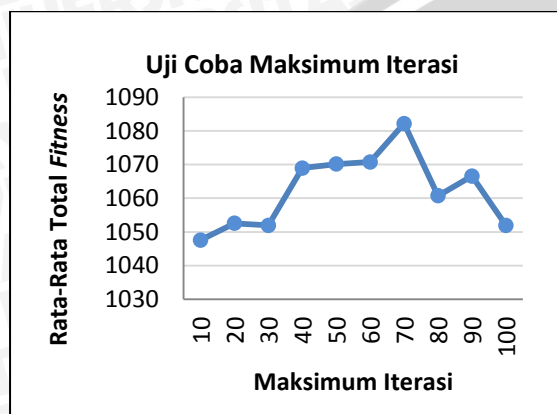
Pada penelitian ini, pengujian dilakukan untuk mengetahui parameter-parameter PSO yang paling optimal untuk menghasilkan nilai total *fitness* terbaik, yaitu meliputi :

1. Pengujian maksimum iterasi

Hasil pengujian maksimum iterasi dapat digambarkan dengan grafik seperti Gambar 3. Maksimum iterasi yang diuji adalah iterasi

kelipatan 10, dimulai dari 10 sampai 100, masing-masing maksimum iterasi diuji sebanyak 5 kali. Secara detail parameter-parameter PSO yang digunakan untuk uji coba maksimum iterasi adalah sebagai berikut :

Jumlah partikel	= 45
Maksimum populasi	= 10
$\theta_{min}, \theta_{max}$	= 0.4, 0.9
c_{1i}, c_{1f}	= 2.5, 0.5
c_{2i}, c_{2f}	= 0.5, 2.5
r_1, r_2	= 0.5, 0.5



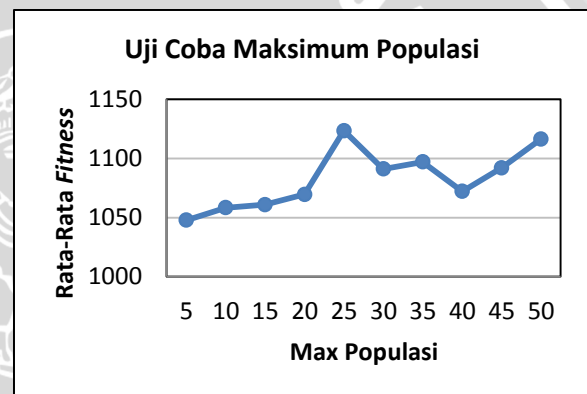
Gambar 3. Grafik Hasil Uji Coba Maksimum Iterasi

Pada kebanyakan kasus, sering ditemukan bahwa dengan bertambahnya iterasi yang dilakukan maka nilai *fitness* yang dihasilkan akan semakin baik yang berarti semakin banyak iterasi yang dilakukan maka kemungkinan untuk mendapatkan nilai terbaik juga semakin besar karena proses *update* posisi yang lebih sering, seperti yang ditunjukkan pada maksimum iterasi sebanyak 10 sampai 70 pada Gambar 3, grafik rata-rata total *fitness* cenderung mengalami kenaikan. Namun, kelemahan untuk iterasi yang banyak adalah membutuhkan waktu komputasi yang lebih lama, akan terasa sia-sia jika sebelum mencapai batas maksimum iterasi sudah mendapatkan nilai *fitness* yang konvergen, dimana nilai yang konvergen menandakan posisi partikel tidak berubah karena sudah dianggap menemukan solusi optimal, terlebih lagi solusi yang dianggap optimal tersebut belum pasti merupakan solusi optimal sesuai yang diharapkan. Jadi, iterasi yang banyak tidak selalu memberikan nilai yang bagus seperti yang ditunjukkan pada Gambar 6.1 dimana iterasi yang lebih besar dari 70 tidak menghasilkan nilai rata-rata total *fitness* yang lebih baik atau sama, bahkan cenderung menurun. Sehingga, jumlah maksimum iterasi yang digunakan pada penelitian ini adalah sebanyak 70 iterasi.

2. Pengujian maksimum populasi

Hasil pengujian maksimum populasi dapat digambarkan dengan grafik seperti Gambar 4. Maksimum populasi yang diuji adalah maksimum populasi dengan kelipatan 5, dimulai dari 5 iterasi sampai 50 iterasi. Maksimum iterasi yang digunakan dalam menguji maksimum populasi adalah 70 iterasi, dimana iterasi sebanyak 70 kali merupakan maksimum iterasi paling optimal yang didapatkan dari uji coba sebelumnya. Secara detail, parameter-parameter PSO yang digunakan untuk uji coba maksimum populasi adalah sebagai berikut:

Jumlah partikel	= 45
Max iterasi	= 70
$\theta_{min}, \theta_{max}$	= 0.4, 0.9
c_{1i}, c_{1f}	= 2.5, 0.5
c_{2i}, c_{2f}	= 0.5, 2.5
r_1, r_2	= 0.5, 0.5



Gambar 4. Grafik Hasil Uji Coba Maksimum Populasi

Sama halnya dengan maksimum iterasi, maksimum populasi juga berpengaruh terhadap nilai *fitness*. Semakin banyak populasi berarti semakin banyak juga partikel yang digunakan, semakin banyak partikel yang digunakan maka semakin baik solusi yang didapatkan. Partikel yang banyak berarti akan menyediakan pilihan posisi partikel yang banyak juga, sehingga besar kemungkinan untuk mendapatkan solusi optimal karena adanya ruang pencarian yang lebih besar untuk dijelajahi tiap iterasinya. Populasi yang banyak pasti akan membutuhkan waktu komputasi yang lebih lama (Engleberth, 2007). Terkadang komputasi yang lama tidak sebanding dengan peningkatan nilai *fitness* yang didapatkan seperti yang terlihat pada Gambar 4, bahwa rata-rata total *fitness* terbesar ketika ditentukan populasi sebanyak 20 sedangkan untuk populasi yang lebih besar tidak didapatkan nilai rata-rata total *fitness* yang lebih baik dari populasi sebanyak 20. Pada umumnya populasi dengan partikel sebanyak 20 sampai 40 sudah mampu mendapatkan nilai yang cukup baik. Jumlah partikel ditentukan sesuai dengan kompleksitas masalah yang dihadapi. Pada uji coba

ini didapatkan banyak populasi yang paling optimal adalah 25 karena menghasilkan nilai rata-rata total fitness terbesar yaitu 1123.547. Sehingga, penggunaan 25 populasi dirasa cukup optimal untuk permasalahan optimasi yang dihadapi dan akan lebih menghemat waktu dari pada banyak populasi yang lebih besar lagi.

3. Pengujian kombinasi nilai minimum (θ_{min}) dan nilai maksimum (θ_{max}) bobot inersia

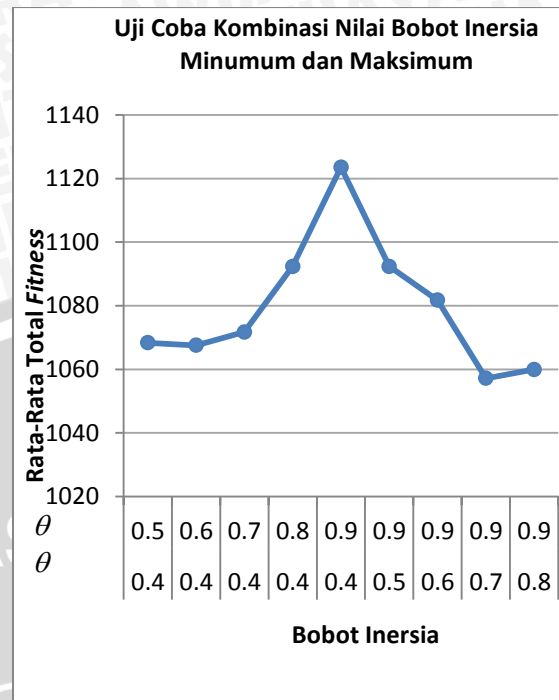
Hasil pengujian Pengujian kombinasi nilai minimum (θ_{min}) dan nilai maksimum (θ_{max}) bobot inersia dapat digambarkan dengan grafik seperti Gambar 5. Dalam pengujian ini digunakan maksimum iterasi = 70 dan maksimum populasi adalah 25, karena kedua nilai tersebut terbukti optimal pada pengujian sebelumnya, sedangkan kombinasi nilai θ_{max} dan θ_{min} yang digunakan dalam pengujian ini yang dapat dilihat pada Tabel 1.

Tabel 1 kombinasi Nilai Bobot Inersia Minimum dan Maksimum

θ_{min}	θ_{max}
0.4	0.5
0.4	0.6
0.4	0.7
0.4	0.8
0.4	0.9
0.5	0.9
0.6	0.9
0.7	0.9
0.8	0.9

Detail nilai parameter PSO yang digunakan adalah sebagai berikut :

Jumlah partikel	= 45
Maksimum iterasi	= 70
Maksimum Populasi	= 25
c_{1i}, c_{1f}	= 2.5, 0.5
c_{2i}, c_{2f}	= 0.5, 2.5
r_1, r_2	= 0.5, 0.5



Gambar 5. Grafik Hasil Uji Coba Kombinasi Nilai Bobot Inersia Minimum dan Maksimum

Pada Gambar 5 dapat dilihat bahwa grafik terus meningkat seiring bertambahnya selisih atau jarak antara nilai θ_{max} dan θ_{min} dan sebaliknya grafik terus menurun seiring berkurangnya selisih nilai θ_{max} dan θ_{min} . Ketika jarak antara nilai θ_{max} dan θ_{min} semakin besar maka jarak nilai penurunan kecepatan dari iterasi ke iterasi yang lain akan juga semakin besar yang berarti kecepatan partikel semakin diperlambat. Kecepatan yang semakin diperlambat akan memberikan kesempatan eksploitasi lokal lebih besar. Eksploitasi akan berguna untuk mencari solusi optimal pada suatu wilayah sebelum melakukan eksplorasi ke wilayah lain, namun jika kesempatan eksploitasi terlalu kecil maka partikel akan cenderung melakukan eksplorasi ke wilayah baru dan kehilangan kesempatan eksploitasi lebih dalam terhadap wilayah-wilayah yang dikunjungi akibatnya solusi optimal pada suatu wilayah tertentu sering terlewatkan (Novitasari, 2015). Pada hasil uji coba dapat dibuktikan bahwa nilai $\theta_{max} = 0.9$ dan $\theta_{min} = 0.4$ menghasilkan rata-rata total fitness terbesar karena jarak/selisih antara kedua nilai tersebut paling besar daripada nilai kombinasi θ_{max} dan θ_{min} lain. Pada penelitian lain juga telah dibuktikan bahwa dengan penentuan nilai $\theta_{max} = 0.9$ dan $\theta_{min} = 0.4$ dapat menghasilkan solusi yang lebih baik untuk permasalahan yang sedang dihadapi (Ratnaweera, Halgamuge, & Watson, 2004).

4. Pengujian kombinasi Koefisien akselerasi yaitu $c_{1i}, c_{1f}, c_{2i}, c_{2f}$

Hasil pengujian kombinasi Koefisien akselerasi yaitu dapat digambarkan dengan grafik seperti Gambar 6. Nilai yang di uji untuk kombanasi ke empat nilai tersebut dapat dilihat pada Tabel 6.5. Untuk maksimum iterasi yang digunakan adalah 70, maksimum populasi adalah 25, $\theta_{max} = 0.9$ dan $\theta_{min} = 0.4$, karena nilai-nilai tersebut terbukti optimal pada pengujian sebelumnya.

Tabel 2. Kombinasi Nilai Koefisien Akselerasi

c_1		c_2	
c_{1i}	c_{1f}	c_{2i}	c_{2f}
2	1	1	2
2.25	0.5	0.5	2.25
2.5	0.5	0.5	2.5
2.5	0.75	0.75	2.5

Sumber : (Ratnaweera, Halgamuge, & Watson, 2000

Detail nilai parameter PSO yang digunakan adalah sebagai berikut :

Jumlah partikel = 45

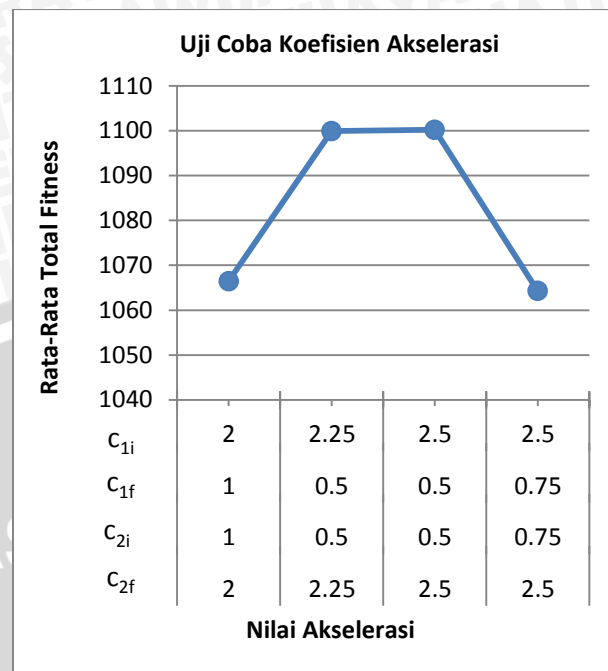
Maksimum iterasi = 70

Maksimum Populasi = 25

$\theta_{min}, \theta_{max}$ = 0.4, 0.9

r_1, r_2 = 0.5, 0.5

Berdasarkan Gambar 6 , nilai rata-rata total *fitness* cenderung besar saat jarak /selisih nilai antar koefisien akselerasi mendekatai atau bernilai 2. dapat dilihat bahwa nilai rata-rata total *fitness* yang terbaik didapatkan ketika nilai koefisien c_{1i}, c_{1f} = 2.5, 0.5 dan koefisien c_{2i}, c_{2f} = 0.5, 2.5 yaitu sebesar 1100.2398, dimana selisih antar koefisiennya bernilai 2. Pada umumnya, dari *range* 1.5-2 untuk c_1 dan dari *range* 2-2.5 untuk c_2 dipilih nilai 2 pada keduanya, karena dapat menghasilkan solusi yang lebih optimal. Nilai 2 dianggap cukup karena tidak terlalu kecil dan tidak terlalu besar karena apabila c_1 dan c_2 yang terlalu besar membutuhkan waktu yang lebih lama untuk mendapatkan solusi optimal karena konvergensi akan lebih lama didapatkan, sedangkan untuk nilai c_1 dan c_2 yang semakin kecil akan mengurangi kemungkinan mendapat solusi paling optimal karena terjadinya konvergensi dini. Pada uji coba penelitian lain juga didiapatkan bahwa saat menggunakan nilai c_{1i}, c_{1f} = 2.5, 0.5 dan c_{2i}, c_{2f} = 0.5, 2.5 dapat menghasilkan nilai rata-rata *fitness* yang lebih baik dengan standar devisiasi yang kecil (Ratnaweera, Halgamuge & Watson, 2004).



Gambar 6. Grafik Uji Coba Kobinasi Koefisien Akselerasi

5. KESIMPULAN DAN SARAN

Kesimpulan

Berdasarkan Penelitian dan hasil pengujian didapat beberapa kesimpulan sebagai berikut :

1. Implementasi algoritma untuk sisitem optimasi pemerataan mutasi Guru SMP di Kabutaen Lumajang terdiri dari 5 tahapan utama yaitu inisialisasi partikel, Perhitungan nilai *fitness*, Penentuan P_{best} dan G_{best} , *update* kecepatan, dan *update* posisi. Kondisi pemeberhentian proses pencarian berdasarkan maksimum iterasi.
2. Uji coba yang dilakukan meliputi uji coba terhadap maksimum populasi, maksimum iterasi, kombinasi nilai bobot inersia minimum dan maksimum , serta pengujian terhadap kombinasi koefisien akselerasi.
3. Dari hasil pengujian didapatkan parameter-parameter PSO yang paling optimal diantaranya untuk maksimum iterasi=70, maksimum populasi =25, $\theta_{max} = 0.9$, $\theta_{min} = 0.4$, $c_{1i}, c_{1f} = 2.5, 0.5$ dan $c_{2i}, c_{2f} = 0.5, 2.5$

Saran

Saran untuk pengembangan lebih lanjut agar sistem optimasi pemerataan mutasi guru dapat menghasilkan solusi pemerataan yang lebih optimal dan memiliki performa yang lebih baik , antara lain:

1. Menggunakan lebih banyak paremeter mutasi agar hasil optimasi lebih kuat. Sehingga

2. Pencarian jarak antara tempat tinggal guru ke masing-masing guru menggunakan data alamat yang sudah dikonversi menjadi data longitude dan latitude agar lebih akurat.
3. Sistem optimasi pemerataan mutasi guru bisa dikembangkan untuk mutasi guru di jenjang pendidikan lain.
4. Sistem optimasi pemerataan mutasi guru bisa dikembangkan dengan menggunakan metode PSO jenis lainnya.

6. DAFTAR PUSTAKA

- Akuntono I., 2011. [online] Tersedia di <http://edukasi.kompas.com/read/2011/11/29/10143063/SKB.5.Menteri.Fokus.Kelola.Distribusi.Guru> [Diakses 12 Desember 2015].
- Engleberth AP., 2007. *Computational Intelligent: An introduction 2nd ed.*, West Sussex: John Wiley & Sons Ltd
- Mahmudy, WF., 2015, *Improved Particle Swarm Optimization Untuk Menyelesaikan Permasalahan Part Type Selection Dan*
- Machine Loading Pada Flexible Manufacturing System (FMS)*. Konferensi Nasional, Sistem Informasi, Universitas Klabat, Airmadidi, Minahasa Utara, Sulawesi Utara.
- Kementerian Riset, Teknologi, dan Pendidikan Tinggi, 2011. Petunjuk Teknis Pelaksanaan Peraturan Bersama5 menteri tentang Penataan dan Pemerataan Guru PNS. [pdf] Kementerian Riset, Teknologi, dan Pendidikan Tinggi. Tersedia di : <http://www.kopertis12.or.id/wp-content/uploads/2012/01/juknis-peraturan-bersama-lima-menteri-tentang-penataan-pemerataan-guru-pns> . [Diakses 12 Desember 2015]
- Ratnaweera A., Halgamuge, SK., Watson HC., 2004. Self Organizing Hierarchical Particle Swarm Optimizer With Time-Varying Acceleration Coefficients. IEEE.
- Santosa B., Willy P., 2011. Metoda Metaheuristik, Konsep dan Implementasi. Graha Ilmu. Surabaya.

