

Deployment Jaringan Sensor Nirkabel berdasarkan Algoritma Particle Swarm Optimization

Zawiyah Saharuna¹, Widyawan², Sujoko Sumaryono³.

Pervasive and Mobile Computing Group

Jurusan Teknik Elektro dan Teknologi Informasi UGM

Jln. Grafika 2 Fakultas Teknik Yogyakarta 55281, Indonesia

wia_saharuna_s210@mail.te.ugm.ac.id, widyawan@ugm.ac.id, sujoko.s@gmail.com

Abstract-Deployment is one of several important issues in Wireless Sensor Network (WSN). During WSN deployment, the connectivity between each sensor nodes must be considered carefully to create reliable communication. In this research, we propose a WSN deployment tool based on Particle Swarm Optimization (PSO) algorithm with connectivity of the wireless to be our concern. Implementation of the PSO algorithm is focused to optimize received power of each sensor node based on its position in the 2D space. Therefore, every sensor node in the network will be able to reach its best position and improves the network connectivity. The deployment results with various transmit power (i.e. -1 dBm, -4 dBm, -7 dBm, -10 dBm, -13 dbm) are successfully form a full-mesh network with well maintained connectivity. However, the use of transmission power higher or lower than the set value requires several adjustments in the input parameter.

Keywords-WSN; PSO; Deployment.

I. PENDAHULUAN

Wireless Sensor Network (WSN) atau jaringan sensor nirkabel merupakan salah satu teknologi inti dalam bidang *Ubiquitous Computing*. WSN terdiri atas satu atau lebih *node sensor* yang digunakan untuk menangkap informasi sesuai dengan karakteristiknya[1]. Teknologi WSN dapat diterapkan di berbagai area karena didukung oleh bentuk perangkat keras yang kecil serta media komunikasi yang menggunakan jaringan nirkabel sehingga bisa berinteraksi dengan baik dalam lingkungan.

Kondisi fisik lingkungan diterapkannya WSN akan mempengaruhi tingkat konektivitas dari *node sensor* dalam jaringannya. Padahal konektivitas merupakan penentu keberhasilan dalam komunikasi antar *node sensor*. Oleh karena itu perlu memperhatikan posisi *node sensor* ketika proses *deployment* dalam penerapan WSN. Beberapa peneliti telah mengajukan berbagai algoritma untuk mengoptimalkan proses *deployment* di lingkungan sehingga kinerja WSN bisa meningkat.

Lee menggunakan *basic deployment algorithm* untuk proses *deployment node* secara otomatis berdasarkan algoritma *Swarm Intelligence* kemudian mengoptimasi beberapa parameter seperti lebar area cakupan dan jumlah *overlapping* [2]. Dong Li, dkk menghadirkan sebuah aplikasi optimasi proses *deployment* menggunakan algoritma *Deployment Technology based on Ant Colony Optimization* (DT-ACO) dengan mempertimbangkan konektivitas jaringan nirkabel, area

cakupan sensor, dan biaya. Mereka juga menawarkan *Power-Aware Cross layer Scheme* (PACS) untuk menambah waktu hidup dan keakuratan pengamatan bagi *node sensor* [3]. Selain itu, *Memetic Simulated annealing* juga digunakan sebagai solusi atas permasalahan dalam proses *deployment* untuk optimasi area cakupan dan waktu hidup dari *node sensor* [4].

Beberapa penelitian juga mencoba menawarkan algoritma baru seperti algoritma *robot-deployment* untuk mengatasi penghalang yang tidak diperkirakan dan mengoptimalkan area sebaran bagi *node sensor* dalam jumlah minimal [5]. Serta algoritma *Obstacle-Resistant Robot Deployment* (ORRD) yang melibatkan kebijakan desain penempatan *node*, kebijakan *serpentinemovement*, keberadaan penghalang, dan batasan area. Algoritma ini mampu dengan cepat menyebarkan *node sensor* dalam jumlah minimal untuk mencakup *sensing area* secara penuh, meskipun terdapat penghalang yang tidak diperkirakan dengan bentuk teratur atau tidak teratur [6].

Pada penelitian ini diajukan suatu aplikasi untuk proses *deployment* berdasarkan Algoritma *Particle Swarm Optimization* (PSO) dengan mempertimbangkan konektivitas jaringan nirkabel. Algoritma PSO dipilih berdasarkan keunggulannya, yaitu mudah diimplementasikan karena persamaan matematisnya sederhana dan hanya memiliki sedikit fungsi operasi dan parameter yang harus ditentukan [7]. Hal inilah yang menjadikan algoritma PSO memiliki karakteristik cepat dalam eksekusi.

II. MODEL PROPAGASI GELOMBANG DI RUANG BEBAS

Propagasi ruang bebas terjadi bila ada jalur *Line of Sight* (LoS) yaitu tidak ada penghalang yang menghalangi propagasi gelombang radio antara pemancar dan penerima. Pada propagasi ruang bebas hanya ada rugi-rugi sinyal sebagai fungsi jarak antara pemancar dan penerima. Kerusakan sinyal hanya dipengaruhi oleh rancangan antena pemancar dan penerima, pola radiasi antenanya, dan lain-lain. Propagasi ruang bebas tidak dipengaruhi oleh distorsi jalur propagasi. Model yang paling sederhana adalah apabila kondisi saling melihat antara pemancar dan penerima terpenuhi dan hanya ada satu sinyal langsung yang diterima, sehingga perhitungan redaman dilakukan dengan menggunakan rumus redaman ruang bebas (free space loss). Persamaan (1) adalah rumus redaman ruang bebas dalam *decibel* [8].

$$L = 32,44 + 20 \log f + 20 \log d \quad (1)$$

L : redaman ruang bebas.
 f : frekuensi dalam MHz.
 d : jarak antar pengirim-penerima dalam Km.

Besarnya kekuatan sinyal yang diterima pada penerima dapat dihitung berdasarkan rumus berikut:

$$Pr = Pt + Gt + Gr - L \quad (2)$$

$$Pr = Pt + Gt + Gr - (32,44 + 20 \log f + 20 \log d) \quad (3)$$

Pt : daya yang dipancarkan oleh pengirim.
 Gt : penguatan antenna pemancar/pengirim.
 Gr : penguatan antenna penerima.

III. PARTICLE SWARM OPTIMIZATION (PSO) DAN IMPLEMENTASI

Algoritma *Particle Swarm Optimization* (PSO) merupakan salah satu algoritma optimasi berbasis populasi yang terinspirasi oleh perilaku sosial binatang seperti kawanan burung atau ikan. Algoritma ini diperkenalkan oleh Eberhart dan Kennedy pada tahun 1995. Berbeda dengan teknik optimasi lainnya, setiap partikel dalam PSO berhubungan dengan suatu *velocity*. Partikel tersebut bergerak melalui penelusuran ruang dengan *velocity* yang dinamis disesuaikan menurut perilaku historisnya. Oleh karena itu partikel-partikel mempunyai kecenderungan untuk bergerak ke area penelusuran yang lebih baik setelah melewati proses penelusuran[9].

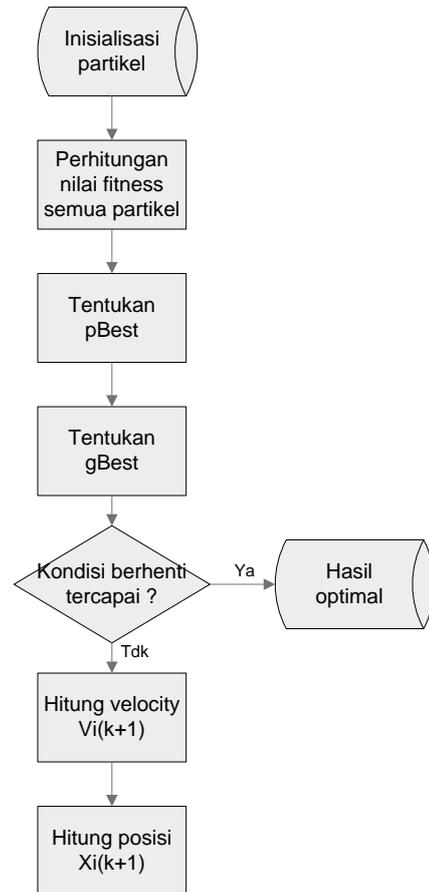
PSO dimulai dengan sekumpulan partikel (solusi) yang dibangkitkan secara acak. Kemudian setiap partikel dievaluasi kualitasnya menggunakan fungsi *fitness*. Selanjutnya, partikel-partikel akan terbang mengikuti partikel yang optimum. Pada setiap generasi (iterasi), setiap partikel diperbaharui berdasarkan dua nilai terbaik yaitu *fitness* terbaik yang dicapai oleh satu partikel saat ini yang dikenal dengan istilah *personal best* (pBest) dan *fitness* terbaik yang dicapai dari semua partikel yang ada pada suatu kawanan yang disebut *global best* (gBest). Setelah menemukan dua nilai terbaik tersebut maka setiap partikel *i* pada posisi X_i memperbaharui vektor *velocity* dan posisinya berdasarkan persamaan berikut:

$$V_i^{k+1} = \omega \cdot V_i^k + n1 \cdot r1(pBi^k - X_i^k) + n2 \cdot r2(gBi^k - X_i^k) \quad (4)$$

$$X_i^{k+1} = X_i^k + V_i^{k+1} \quad (5)$$

V_i^{k+1} : *velocity* partikel *i* pada iterasi $k+1$.
 X_i^k : partikel *i* pada iterasi k .
 pBi^k : *personal best* partikel *i* pada iterasi k .
 gBi^k : *global best* partikel *i* pada iterasi k .
 $n1, n2$: nilai laju pembelajaran.
 $r1, r2$: nilai acak dari 0 sampai 1.
 ω : bobot inersia

Prosedur yang digunakan untuk menerapkan algoritma PSO dapat dilihat pada Gambar 1.



Gambar 1. Diagram alir PSO

Implementasi dari PSO pada proses *deployment* adalah sebagai berikut:

A. Ukuran Swarm

Ukuran *swarm* atau populasi yang dipilih adalah tergantung pada persoalan yang dihadapi. Ukuran *swarm* yang umum digunakan berkisar antara 20 sampai 50, meski tidak menutup kemungkinan untuk menggunakan ukuran *swarm* yang lebih besar lagi. Carlisle menyatakan bahwa ukuran *swarm* akan mempengaruhi kecepatan proses. Ukuran *swarm* yang terlalu kecil bisa terjebak pada optimum lokal meskipun waktu prosesnya sangat cepat. Sebaliknya, ukuran *swarm* yang besar jarang terjebak pada optimum lokal tetapi waktu prosesnya lebih lama[10]. Pada penelitian ini digunakan 30 partikel dengan pertimbangan efisiensi waktu dan pencapaian solusi yang mendekati optimum global.

B. Representasi Partikel

Partikel (X_i) merupakan suatu solusi yang dibangkitkan secara acak dan selanjutnya dioptimasi untuk menghasilkan solusi yang baik. Pada penelitian ini fokus optimasi adalah pada daya yang diterima oleh masing-masing *node sensor* ketika disebar berdasarkan posisi *node sensor*, sehingga partikel yang digunakan merepresentasikan posisi dari sejumlah *node sensor* dalam koordinat 2D dengan area *deployment* berbentuk bujur sangkar. Jumlah *node sensor* yang akan disebar bervariasi disesuaikan dengan kebutuhan pengguna aplikasi. Representasi dari partikel dapat dilihat pada Gambar 2. Untuk menjaga *deployment* dari *node sensor* tetap berada

dalam area *deployment* maka nilai $X_{i,j}$ dibatasi pada range tertentu yaitu batas bawah ($X_a = 0,0$) dan batas atas ($X_b =$ luas area sebaran).

$X_{1,1}$	$X_{1,2}$	\dots	$X_{1,j}$
$X_{2,1}$	$X_{2,2}$	\dots	$X_{2,j}$
\dots	\dots	\dots	\dots
$X_{i,1}$	$X_{i,2}$	\dots	$X_{i,j}$

Gambar 2. Representasi partikel

$X_{i,j}$: posisi partikel i *node sensor* j pada koordinat (x,y)

i : ukuran *swarm*

j : jumlah *node sensor*

C. Fungsi Fitness

Selain representasi solusi, maka komponen penting lainnya dari PSO adalah fungsi *fitness*. Pada perancangan *deployment node sensor* ini digunakan fungsi *fitness* yang ditentukan oleh hal-hal berikut :

1) Konektivitas diukur berdasarkan daya yang diterima oleh sebuah *node*, dalam hal ini batasan maksimal daya yang diterima suatu *node* dianggap terkoneksi dengan *node* yang lain adalah -70 dBm. Nilai batasan yang ditetapkan lebih besar dari batas sensitivitas penerimaan daya pada IQRF tipe TR-52B yaitu -100 dBm karena pertimbangan jangkauan frekuensi radio TR-52B bisa mencapai 700 m (1.2 kb/s) dan 500 m (19.2 kb/s), padahal luas area sebaran yang ditetapkan dalam pengujian tidak seluas area tersebut [11].

2) Skenario proses *deployment* dengan konfigurasi *full-mesh* yaitu semua *node sensor* dalam jaringan saling terkoneksi. Kondisi terbaik ketika jarak antar *node sensor* maksimal tetapi daya yang diterima oleh *node sensor* tersebut tetap lebih besar atau sama dengan -70dBm/-100dB dan jumlah koneksi mendekati $N(N-1)$, dengan N adalah jumlah *node sensor* yang disebar.

Berdasarkan skenario tersebut maka ditetapkan fungsi *fitness* sebagai berikut:

$$F(X_i) = \sum_0^{i-1} Pr_b(X_{ij}) + \sum_0^{i-1} A(X_{ij}) \quad (6)$$

$F(X_i)$: fungsi *fitness* partikel i

$Pr_b(X_{ij})$: sinyal terbaik yang diterima partikel i *node sensor* j .

$A(X_{ij})$: jumlah *node* tetangga yang terdeteksi pada partikel i *node sensor* j .

$$Pr_b(X_{ij}) = \min[Pr(X_{ij})] > (-100dB) \quad (7)$$

$Pr_b(X_{ij})$ merupakan sinyal terbaik yang diterima oleh partikel i *node sensor* j dengan asumsi bahwa kekuatan sinyal terbaik adalah yang mendekati nilai -100dB. Perhitungan kekuatan sinyal yang diterima menggunakan

persamaan (3) dengan mengabaikan nilai penguatan pada antenna karena kecilnya penguatan yang terjadi pada antenna *node sensor* tipe TR-52B[11]. Sedangkan untuk menghitung jarak dengan rumus *Euclidean distance*, seperti menghitung jarak antar koordinat (x_i,y_i) dengan (x_j,y_j) pada persamaan (8).

$$D(i, j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (8)$$

D. Laju Pembelajaran

Pada umumnya nilai-nilai untuk laju pembelajaran $n1$ dan $n2 = 2.0$ dan berada pada rentang antara 0 sampai 4. Pada penelitian ini digunakan nilai $n1$ dan $n2 = 2.0$ dengan pertimbangan untuk menyeimbangkan antara *cognitive part* dan *social part* pada PSO.

E. Bobot Inersia

Perubahan *velocity* pada algoritma PSO terdiri atas tiga bagian yaitu *sosial part*, *cognitive part* dan *momentum part*. Ketiga bagian tersebut menentukan keseimbangan antara kemampuan penelusuran *global* dan *local*, karena dapat memberikan performansi yang baik pada PSO. *Inertia weight* diperkenalkan untuk keseimbangan antara kemampuan penelusuran *global* dan *local* seperti pada persamaan (4). Pada penelitian ini parameter *inertia weight* (w) diperoleh dengan menggunakan persamaan berikut:

$$w = w_{\max} - \frac{w_{\max} - w_{\min}}{Iter_{\max}} Iter_i \quad (9)$$

w_{\max} : bobot maksimal.

w_{\min} : bobot minimal.

$Iter_{\max}$: jumlah iterasi maksimum.

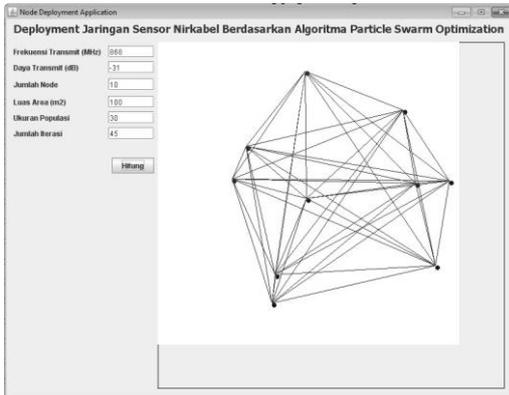
$Iter_i$: iterasi ke i .

IV. HASIL PENGUJIAN APLIKASI

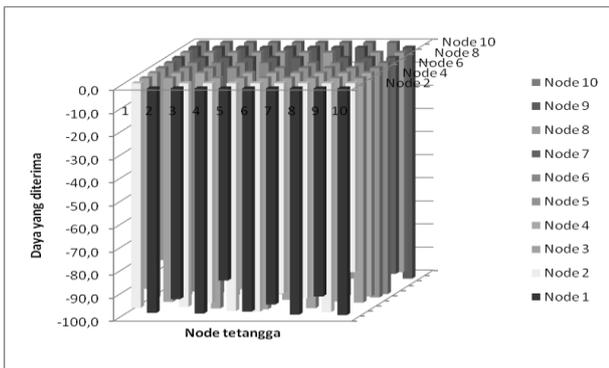
Untuk mengetahui kelayakan implementasi algoritma PSO pada *deployment WSN* dilakukan simulasi pengujian dengan menggunakan bahasa pemrograman Java. Pengujian aplikasi ini menggunakan 10 *node sensor* yang akan disebar pada area dengan ukuran $100 \times 100 m^2$. Komunikasi dalam jaringan menggunakan frekuensi 868 MHz, daya pancar sebesar -31 dB, -34 dB, -37 dB, -40 dB, -43 dB hingga -46 dB yang disesuaikan dengan spesifikasi WSN jenis IQRF tipe TR-52B. Batasan daya yang diterima oleh *node sensor* sehingga dianggap terhubung dengan *node sensor* tetangganya adalah -100dB/-70dBm. Parameter komponen PSO yang digunakan diatur dengan jumlah partikel sebanyak 30 (0-29), nilai laju pembelajaran masing-masing diatur pada nilai 2,0 dengan asumsi *full model* yaitu untuk menyeimbangkan *cognitive part* dan *social part*. Pengujian ini menggunakan *inertia weight* dengan nilai yang dinamis dari setiap variable pada rentang antara 0.4 sampai 0,9 dan iterasi maksimal sebanyak 45 iterasi.

Hasil yang diperoleh dari *deployment* menggunakan daya pancar -31 dB dapat dilihat pada Gambar 3. Hasilnya membentuk hubungan *full-mesh* yaitu semua *node sensor* yang disebar saling terhubung satu sama lain dan konektivitas tetap terjaga seperti yang diinginkan pada skenario desain. Kondisi dua *node sensor* dinyatakan terhubung adalah ketika kedua *node sensor*

tersebut dihubungkan oleh sebuah garis. Garis tersebut merupakan representasi daya $\geq -100\text{dB}$. Sedangkan kondisi tidak terhubung merepresentasikan daya $< -100\text{dB}$.

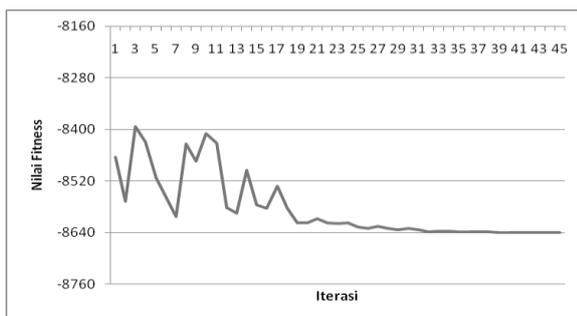


Gambar 3. Hasil deployment



Gambar 4. Grafik penerimaan sinyal setiap node pada partikel ke 23

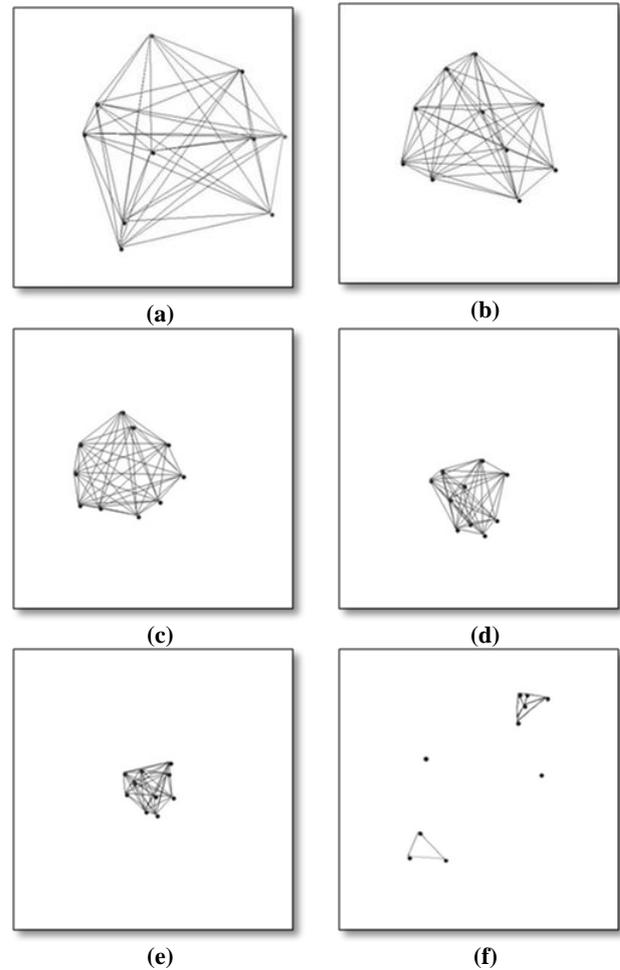
Desain hasil deployment pada Gambar 3. merupakan representasi solusi terbaik dari partikel pada swarm yaitu solusi dari partikel ke 23 yang memiliki nilai fitness terbaik sebesar -8639,3816. Adapun besarnya daya yang diterima oleh masing-masing node sensor pada partikel 23 ketika kondisi konvergen berada dalam jangkauan -81,70 dB sampai dengan -100 dB. Nilai tersebut masih memenuhi batas yang disyaratkan untuk konektivitas. Untuk lebih jelasnya rentang nilai daya yang diterima dapat dilihat pada Gambar 4. Sumbu x menjelaskan node sensor tetangga yang terdeteksi, sedangkan sumbu y adalah nilai daya yang diterima.



Gambar 5. Laju konvergensi

Pada Gambar 5. menjelaskan tentang laju konvergen yang dihasilkan ketika proses deployment 10 node sensor. Hasilnya diperoleh dari nilai fitness terbaik pada

setiap iterasi hingga iterasi maksimal. Kondisi konvergen mulai terjadi pada iterasi ke 31 dengan nilai fitness terbaik -8633.2416 pada indeks partikel ke 12 hingga iterasi maksimal dengan nilai fitness terbaik -8639.3816 pada indeks partikel ke 23.



Gambar 6. Hasil deployment node

Seperti Gambar 3. hasil deployment yang ditunjukkan oleh Gambar 6. juga membentuk hubungan full-mesh yaitu semua node sensor yang tersebar saling terhubung satu sama lain dan tetap mempertahankan konektivitas sehingga memenuhi ketetapan pada skenario desain. Gambar 6(a) sampai dengan 6(e) merupakan hasil deployment node sensor dengan daya pancar secara berurutan sebesar -31 dB, -34 dB, -37 dB, -40 dB, dan -43 dB. Kelima hasil deployment tersebut membentuk jaringan full-mesh dengan kualitas yang berbeda. Semakin besar daya yang dipancarkan maka semakin besar jarak antar node sensor dan kualitas konektivitas jaringannya tetap terjaga. Sebaliknya semakin kecil daya yang dipancarkan maka jarak antar node sensor semakin dekat dan untuk kondisi tertentu kualitas konektivitas jaringannya berkurang. Hasil tersebut juga memenuhi kondisi yang disyaratkan pada penetapan fungsi fitness yang digunakan untuk mengevaluasi nilai fitness partikel. Hasil deployment terbaik diperoleh pada proses deployment menggunakan daya pancar -31dB dengan pencapaian jarak antara dua node sensor maksimal tetapi daya yang diterima oleh node sensor tersebut ≥ -100 dB

dan jumlah koneksi mendekati 90 buah sehingga nilai fungsi *fitness*-nya mendekati nilai *fitness* maksimal -9090. Sedangkan Gambar 6(f) merupakan hasil *deployment* dengan daya pancar yang lebih kecil yaitu -46 dB memiliki konektivitas jaringan nirkabel yang buruk.

Pada Tabel 1. menjelaskan batasan nilai daya yang diterima oleh masing-masing *node sensor* berdasarkan perbedaan daya pancar *node sensor* tetangganya. Ketika daya yang dipancarkan sebesar -31 dB, -34 dB, -37 dB, -40 dB, dan -43 dB daya minimum yang diterima oleh node adalah -100dB, sehingga node tersebut dianggap terhubung dengan tetangganya atau konektivitasnya terjaga. Sedangkan ketika daya pancar berkurang menjadi -46 dB, daya minimum yang diterima melewati batas -100dB, sehingga beberapa *node sensor* tidak terhubung dalam jaringan atau dengan kata lain konektivitasnya buruk.

TABLE I. DAYA YANG DITERIMA

Daya Pancar (dB)	Daya yang Diterima (dB)	
	Maksimal	Minimal
-31	-81,70	-100
-34	-86,55	-100
-37	-84,85	-100
-40	-86,18	-100
-43	-86,87	-100
-47	-84,96	-114,79

Permasalahan yang didapatkan dalam pengujian adalah ketika daya pancar yang digunakan lebih besar dari -31dB dalam hal ini -28dB dan -25dB. Hasil *deployment* untuk daya pancar tersebut membuat *node sensor* keluar dari area sebaran, hal ini disebabkan karena luas area yang digunakan terlalu kecil untuk menampung jaringan sensor nirkabel dengan 10 *node sensor* yang daya pancarnya -28dB dan -25dB. Penyelesaian permasalahan tersebut bisa dengan memperbesar area *deployment*, atau mengurangi jumlah *node sensor* yang akan disebar. Kondisi tersebut menjadi kelemahan dari algoritma yang diajukan karena luas area dan letak *node sensor* dalam koordinat 2D tidak dimasukkan dalam parameter yang akan dioptimasi.

V. KESIMPULAN

Algoritma PSO cukup baik digunakan pada *deployment* jaringan sensor nirkabel seperti algoritma jenis *swarm intelligence* lainnya. Pengujian *deployment* WSN jenis IQRF tipe TR-52B dengan enam daya pancar yang berbeda mendapatkan hasil bahwa semakin besar daya yang dipancarkan maka semakin besar jarak antar

node sensor dan kualitas konektivitas jaringannya tetap terjaga. Sebaliknya semakin kecil daya yang dipancarkan maka jarak antar *node sensor* semakin dekat dan untuk kondisi tertentu kualitas konektivitas jaringannya akan berkurang. Meski demikian, Algoritma ini memiliki kekurangan ketika daya pancar yang digunakan lebih besar dari -31dB diperoleh hasil *deployment* yang buruk karena *deployment node sensor* melebihi batas area. Kondisi tersebut merupakan kelemahan dari algoritma ini karena luas area sebaran dan letak *node sensor* dalam koordinat 2D tidak masuk dalam parameter yang dioptimasi. Hal ini bisa dijadikan pertimbangan untuk penelitian selanjutnya.

REFERENCES

- [1] Akyildiz I.F., W. Su, Sankarasubramaniam Y., Cayirci E, "Wireless Sensor Networks: a Survey," Computer Networks 38: 393-422. 2002.
- [2] KwangEui Lee, "An Automated Sensor Deployment Algorithm Based on Swarm Intelligence for Ubiquitous Environment," International Journal of Computer Science and Network Security 7: No.12. 2007.
- [3] Dong Li, Wei Liu, Ze Zhao, Li Cui. 2008, "Demonstration of A Wsn Application in Relic Protection and an Optimized System Deployment Tool," IEEE Information Processing in Sensor Networks: 541-542, April 2008 [International Conference; St. Louis, MO, 22-24 Apr 2008].
- [4] Molina G dan Alba E, "Wireless Sensor Network Deployment Using a Memetic Simulated Annealing," IEEE Applications and the Internet: 237-240, 2008 [International Symposium; Turku, 28 Jul 2008-1 Ags 2008].
- [5] Chang Chih-Yung, Sheu Jang-Ping, Chen Yu-Chieh, dan Chang Sheng-Wen, "An Obstacle-Free and Power-Efficient Deployment Algorithm for Wireless Sensor Network," IEEE Systems, Man and Cybernetics, Part A: Systems and Humans 39 Issue: 4: 795 - 806, 2009.
- [6] Chang Chih-Yung, Chang Chao-Tsun, Chen Yu-Chieh, Chang Hsu-Ruey, "Obstacle-Resistant Deployment Algorithm for Wireless Sensor Networks," IEEE Vehicular Technology 58: 6: 2925-2941, 2009.
- [7] Haupt RL dan Haupt SE., Practical Genetic Algorithm, 2nd Ed. Hoboken, New Jersey: John Wiley & Sons, Inc. 2004.
- [8] C. Balanis, Antenna theory - analysis and design, 2nd Ed. John Wiley & Sons, 1997.
- [9] Schutte JF., The Particle Swarm Optimization Algorithm, EGM 6365-Structural Optimization.. http://www.mae.ufl.edu/haftka/stropt/Lectures/PSO_introduction.pdf [19 apr 2011]. 2005.
- [10] Carlisle A. and Dozier G., "An Off-The-Shelf PSO," Proceedings of the Particle Swarm Optimization, Workshop, 2001, pp 1-6.
- [11] datasheet IQRF TR52-B, <http://www.iqrf.org/weben/downloads.php?id=91>.
- [12] Shi, Y. H. and Eberhart, R.C., "Parameter Selection in Particle Swarm Optimization," Evolutionary Programming VII: Proc. EP 98 pp. 591-600, Springer-Verlag, New York, 1998.
- [13] Shi, Y. H. and Eberhart, R.C., "A Modified Particle Swarm Optimizer," IEEE International Conference on Evolutionary Computation, Anchorage, Alaska, May 4-9, 1998.
- [14] Karl Holger dan Willig Andreas, Protocols and Architectures for Wireless Sensor Network, England: John Wiley & Sons, Ltd. 2005.