

PERANCANGAN LALUAN BAGI ROBOT MUDAH ALIH MENGUNAKAN ALGORITMA KOLONI LEBAH BUATAN

Muhamed Faiz Abdul Kader
Abdul Hadi Abd Rahman

Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia

ABSTRAK

Perancangan laluan yang optimum telah mendapat banyak perhatian kebelakangan ini terutamanya yang melibatkan pergerakan robot mudah alih. Algoritma perancangan laluan yang tersedia bagi robot biasanya kurang efektif dan seringkali terbabas. Bagi robot mudah alih, perancangan laluan memerlukan lebih banyak perhatian yang menyeluruh dan algoritma yang sesuai diperlukan untuk membuat pergerakan yang optimum serta mengelakkan perlanggaran. Projek ini mencadangkan pendekatan alternatif bagi perancangan laluan untuk robot mudah alih menggunakan algoritma berasaskan koloni lebah buatan (ABC). Algoritma ini akan digunakan bagi meningkatkan kemampuan perancangan laluan dan mengoptimumkan pergerakan robot mudah alih. Projek ini akan membandingkan algoritma ABC dengan algoritma klasik yang lain seperti algoritma pengoptimuman kawanan partikel (PSO) untuk mendapatkan perancangan laluan robot mudah alih yang paling optimum. Beberapa persekitaran telah dibangunkan bagi tujuan ujian prestasi secara simulasi. Kedua-dua algoritma telah dihasilkan untuk membuat perancangan laluan yang terbaik serta mengelakkan perlanggaran. Hasil simulasi telah dianalisis dan perbandingan telah dilakukan berdasarkan objektif kajian. Analisis yang dilakukan menunjukkan bahawa algoritma ABC adalah lebih baik dengan hasil perancangan laluan yang memenuhi objektif sebanyak 83% manakala algoritma PSO hanya sebanyak 17%.

1 PENGENALAN

Robot adalah manipulator terkawal yang mampu mengambil keputusan serta melakukan tugas yang kompleks seperti manusia. Robot boleh menjadi sangat penting dalam pelbagai situasi yang agak merbahaya di mana manusia tidak mampu mencapai objektif tertentu kerana lokasi bentuk muka bumi dan persekitaran yang berbahaya. Robot mudah alih adalah sistem automatik yang mampu bergerak. Robot mudah alih mempunyai kemampuan untuk bergerak di persekitarannya dan tidak terpaku pada satu lokasi fizikal.

Walau bagaimanapun, sistem robotik mungkin terdiri daripada sensor sederhana, memperoleh dan memproses data dan berinteraksi dengan persekitaran yang kompleks. Robot mudah alih telah digunakan secara meluas dalam bidang keselamatan, perindustrian dan penerokaan angkasa lepas kerana mempunyai ciri-ciri seperti ketahanan, boleh diharap dan ekonomi. Perancangan laluan telah menjadi salah satu masalah utama bagi sistem robot (RS).

Perancangan laluan (PP) adalah salah satu prosedur utama dalam RS yang kini banyak digunakan. Ia boleh digambarkan sebagai pencarian laluan yang optimum antara dua kedudukan tanpa bertembung dengan halangan. Pelbagai pendekatan telah dicadangkan untuk menyelesaikan masalah perancangan laluan tetapi penghasilan laluan yang optimum dan bebas dari halangan tetap menjadi cabaran untuk RS kerana sifatnya yang rumit.

Algoritma koloni lebah buatan (ABC) adalah algoritma SI yang dibangunkan baru-baru ini. Algoritma ini diilhamkan oleh tingkah laku koloni lebah yang mencari makan. ABC mempunyai kelebihan pada memori, carian tempatan dan mekanisme penambahbaikan penyelesaian, sehingga dapat mendapat prestasi yang sangat baik pada masalah pengoptimuman dan telah menarik banyak perhatian dalam bidang penyelidikan yang berkaitan.

Pengoptimuman Kawanan Partikel (PSO) pada asalnya dimajukan oleh Eberhart dan Kennedy pada tahun 1995, yang terinspirasi dari metafora interaksi sosial dalam kawanan burung atau sekolah ikan. PSO bermula dengan generalisasi secara rawak populasi penyelesaian berpotensi. Pada masa ini, PSO telah menjadi salah satu algoritma Kecerdasan Kawanan (SI) yang paling popular dan berjaya digunakan untuk banyak bidang penyelidikan, seperti pengoptimuman berangka, masalah jurujual perjalanan, pembinaan rangkaian neural buatan, masalah pengelompokan, masalah penghalaan kenderaan dan sebagainya.

Oleh itu, beberapa persekitaran akan dibangunkan bagi tujuan ujian prestasi secara simulasi. Untuk mengesahkan keberkesanan algoritma yang dicadangkan iaitu ABC, kedua-dua algoritma PSO and ABC akan dikaji untuk mendapatkan prestasi perancangan laluan yang terbaik serta mengelakkan pelanggaran.

2 PENYATAAN MASALAH

Salah satu masalah yang mencabar dalam RS adalah untuk membuat pengiraan laluan yang optimum tanpa perlanggaran serta memperbaiki prestasi sistem semasa menghadapi kekangan. Salah satu cara klasik yang digunakan adalah dengan menghantar robot untuk mengetahui persekitarannya dan menetapkan mercu tanda yang boleh digunakan dalam perancangan laluan. Dalam keadaan ini, robot menggunakan sensornya, peta pembinaan, dan mengemaskini kedudukannya. Walau bagaimanapun, persekitaran terdiri daripada sejumlah besar ketidakpastian dan teknik pemetaan mengambil banyak memori, masa dan tidak mampu menghasilkan laluan yang optimum.

Dalam beberapa dekad yang lalu, beberapa pendekatan klasik telah dibangunkan oleh penyelidik untuk menyelesaikan masalah RS seperti kaedah pemetaan jalan, penguraian sel, penyepunlindapan simulasi, voronoidiagram, penglihatan berdasarkan grafik, rangkaian neural dan sebagainya. Sebilangan besar pendekatan klasik telah diberi perhatian untuk mengoptimumkan jarak laluan. Tetapi, beberapa kelemahan dikaitkan dengan pendekatan tradisional seperti ruang masalah yang lebih besar, melibatkan lebih banyak kerumitan untuk pengiraan dan ada kemungkinan besar penyelesaiannya ruang masalah yang dipertimbangkan mungkin jatuh pada tahap tempatan minimum. Oleh itu, kelemahan ini menyebabkan pendekatan tradisional membazir dalam pelbagai masalah ruangan. Arah arus penyelidikan dalam sistem robot tidak hanya memusatkan perhatian pada kawalan manipulator robot yang tepat tetapi juga memperkasakan mereka dengan kepintaran.

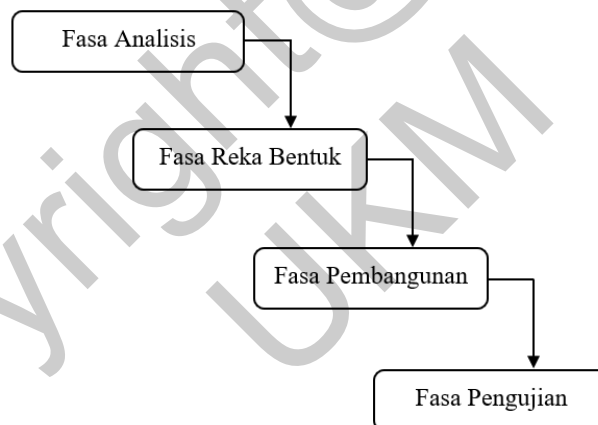
3 OBJEKTIF KAJIAN

Projek ini bertujuan untuk menguji prestasi dua algoritma perancangan laluan bagi robot mudah alih yang terpilih iaitu ABC dan PSO melalui simulasi. Objektif kajian yang utama adalah membandingkan algoritma perancangan laluan serta mengelakkan perlanggaran dari lokasi permulaan ke lokasi yang dituju. Objektif yang kedua ialah menguji jarak laluan bagi robot dari kedudukan awal yang telah ditentukan ke destinasi.

Kertas ini memfokuskan kepada perancangan laluan yang optimum bagi robot mudah alih serta mengelakkan perlanggaran. Oleh itu, algoritma yang efektif adalah sangat diperlukan dan sangat penting untuk menghasilkan perancangan laluan yang optimum. Algoritma yang efektif boleh digunakan untuk bidang-bidang teknologi yang memerlukan penggunaan robot mudah alih.

4 METOD KAJIAN

Dalam kajian, metodologi adalah sangat penting bagi memudahkan seseorang untuk memahami dan jelas tentang projek yang akan dilakukan. Metodologi juga akan melicinkan perjalanan projek supaya lebih teratur dan terarah dalam mencapai objektif. Rajah 1 menunjukkan model metodologi *waterfall* bagi pelaksanaan kajian ini. Terdapat empat fasa utama iaitu fasa analisis, fasa reka bentuk, fasa pembangunan dan fasa pengujian.



Rajah 1 Model metodologi

4.1 Fasa Analisis

Fasa analisis adalah fasa yang paling utama dalam proses membangunkan persekitaran serta algoritma PSO dan ABC untuk digunakan semasa perancangan laluan bagi robot. Tujuan fasa ini adalah untuk mengenal pasti masalah dan skop serta merancang matlamat dan strategi untuk membangunkan projek ini. Selain itu, beberapa kajian lepas telah dirujuk dan dibandingkan untuk mengenal pasti aspek-aspek yang berkaitan dengan projek ini.

Algoritma memainkan peranan penting dalam menetapkan panduan berlangkah supaya permasalahan yang wujud ketika pembangunan dapat dijejak dan diselesaikan mengikut logik panduan. Pseudokod sering digunakan untuk merancang sesuatu algoritma dengan melakar struktur program sebelum memulakan pengekodan yang sebenar. Rajah 2 menunjukkan pseudocode bagi algoritma PSO manakala rajah 3 menunjukkan pseudocode bagi algoritma ABC.

```

1: Memulakan populasi maksimum  $n$ , dimensi  $d$ , leleran maksimum  $k_{max}$ , dan had rendah
dan had tinggi untuk kedudukan dan halaju
2: Memulakan  $x_i$  secara rawak:  $x_{min} \leq x_i \leq x_{max}$ 
3: Memulakan  $v_i$  secara rawak:  $v_{min} \leq v_i \leq v_{max}$ 
4:  $f = p$ 
5:  $g = \min(p)$ 
6:  $k = 1, i = 1$ 
7: while ( $k \leq k_{max}$ ) do
8:   for ( $i = 1 : n$ ) do
9:     Mengira  $v_i$  menggunakan (3)
10:    Mengira  $x_i$  menggunakan (4)
11:    if  $f_i < f_p$  then
12:       $p_i = f_i$ 
13:    end if
14:    if  $f_i < g$  then
15:       $g = f_i$ 
16:    end if
17:  end for
18:   $k = k + 1$ 
19: end while
20: Penyelesaian global optimum =  $g$ 

```

Rajah 2 Pseudocode algoritma PSO

```

1: Memulakan populasi maksimum  $N$ , dimensi  $D$ , leleran maksimum  $k_{max}$ , dan had rendah
dan had tinggi  $x_{min}$ ,  $x_{max}$ 
2: Memulakan  $x_i$  secara rawak:  $x_{min} \leq x_i \leq x_{max}$  and  $i : 1, 2, 3, \dots, N$ 
3: Mengira  $c$  daripada  $x_i$ 
4: Leleran  $k = 1, t = 0, L = 100$ 
5: while ( $k \leq k_{max}$ ) do
6:   Pilih  $s \neq i$  dan  $\varphi_i$  dimana  $-1 \leq \varphi_i \leq 1$ 
7:   Pilih sumber makanan secara rawak  $x_{ij}$  dan menjana makanan baru  $v_{ij}$ 
menggunakan  $s$  and  $\varphi_i$ 
8:    $v_{ij} = x_{ij} + \varphi_{ij} (x_{ij} - x_{sj})$  dimana  $s \in [1, 2, \dots, N]$  dan  $j \in [1, 2, \dots, D]$ 
9:    $f_{vi} = 1 / (1 + |f_i(x_{vi})|)$  if  $c_{vi} \geq 0$ 
10:   $f_{vi} = 1 + |f_i(x_{vi})|$  if  $c_{vi} < 0$ 
11:  if  $f_{vi} > f_{xi}$  then
12:     $v_{ij} = x_{ij}$ 
13:     $g = c_v$ 
14:  end if
15:  if  $f_{vi} < f_{xi}$  then
16:     $t = t + 1$ 
17:  end if
18:  Hitung kebarangkalian untuk penyelesaian baru

$$P_i = \frac{f_i N}{\sum_{i=1}^D f_i}$$

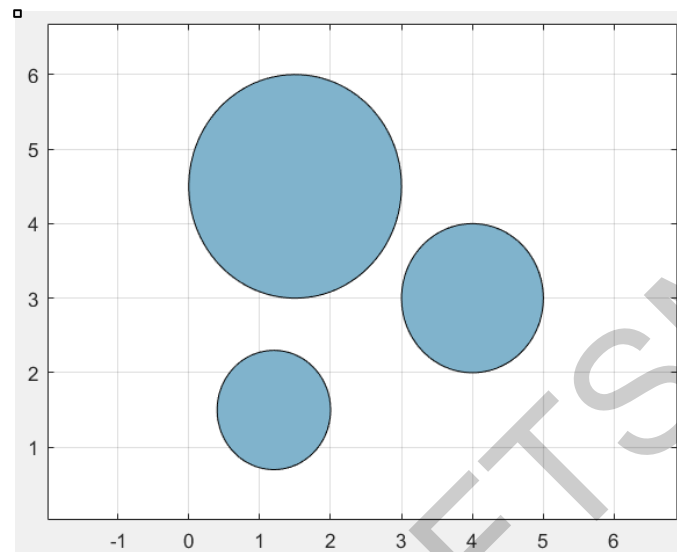
19:  Menghasilkan penyelesaian baru  $v_i$  untuk penonton yang menggunakan nilai  $P_i$  dan
penyelesaian sedia ada  $x_i$ 
20:  Rekodkan  $g$  and  $v_i$ 
21:  Ulang langkah 9 hingga 18 untuk penonton  $v_i$ 
22:  if  $\max(t) \geq L$  then
23:    Hitung kedudukan lebah pengakap
24:     $x_i^j = x_{min}^j + \text{rand}[0, 1](x_{max}^j - x_{min}^j)$ 
25:  end if
26:   $k = k + 1$ 
27: end while
28: Global optimum =  $g$ 

```

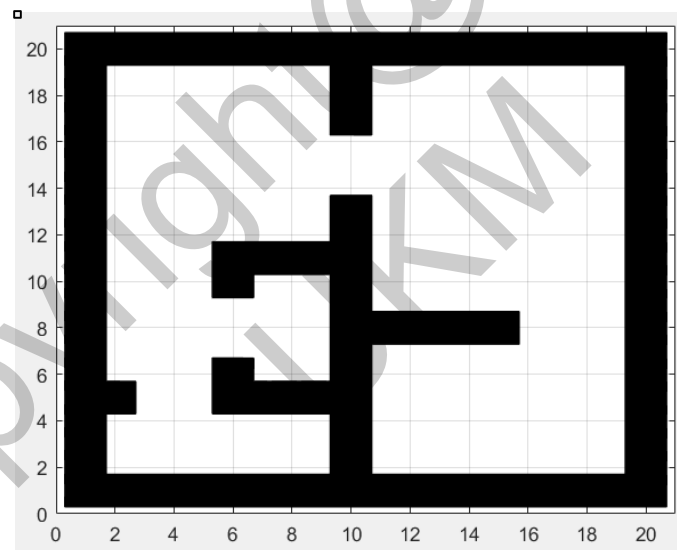
Rajah 3 Pseudocode algoritma ABC

4.2 Fasa Reka Bentuk

Dalam fasa ini, aktiviti-aktiviti utama akan dijalankan seperti menyediakan perisian yang sesuai seperti MATLAB. Persekitaran untuk robot juga dibangunkan untuk menguji perancangan laluan menggunakan dua algoritma yang berbeza. Data-data yang dikumpulkan semasa fasa analisis akan dijadikan sebagai rujukan bagi membangunkan persekitaran tersebut. Tujuan fasa ini adalah untuk memastikan objektif projek dan kajian adalah konsisten serta menentukan bagaimana algoritma tersebut akan berfungsi. Rajah 4 menunjukkan persekitaran dengan halangan bulatan. Halangan bulatan tersebut boleh digambarkan sebagai objek-objek yang berada di dalam kawasan rumah. Rajah 5 menunjukkan persekitaran dengan ruangan bilik yang cukup luas untuk robot membuat pergerakan dari bilik ke bilik. Kedua-dua persekitaran ini dihasilkan menggunakan perisian MATLAB.



Rajah 4 Persekitaran dengan halangan

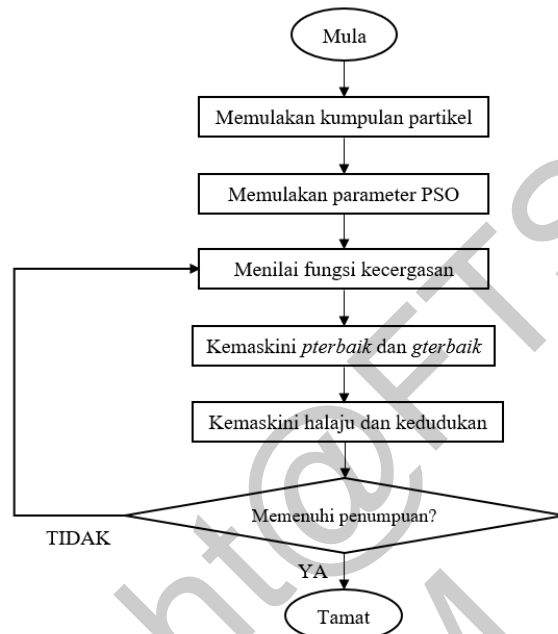


Rajah 5 Persekitaran dengan ruangan bilik

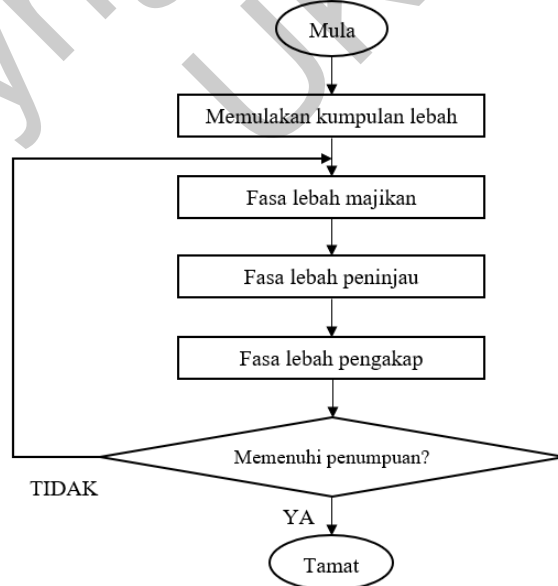
4.3 Fasa Pembangunan

Maklumat-maklumat yang terkumpul pada fasa-fasa sebelumnya akan membantu pada fasa pembangunan. Tujuan fasa ini adalah untuk membangunkan dan menyediakan sistem robot di dalam persekitaran supaya ia boleh berfungsi seperti yang dirancang. Aktiviti-aktiviti utama dalam fasa ini adalah pembangunan algoritma dan perancangan laluan.

Rajah 6 menunjukkan carta alir bagi algoritma PSO dan rajah 7 menunjukkan carta alir bagi algoritma ABC. Rajah tersebut adalah sangat berguna bagi memahami proses kedua-dua algoritma tersebut.



Rajah 6 Carta alir algoritma PSO



Rajah 7 Carta alir algoritma ABC

4.4 Fasa Pengujian

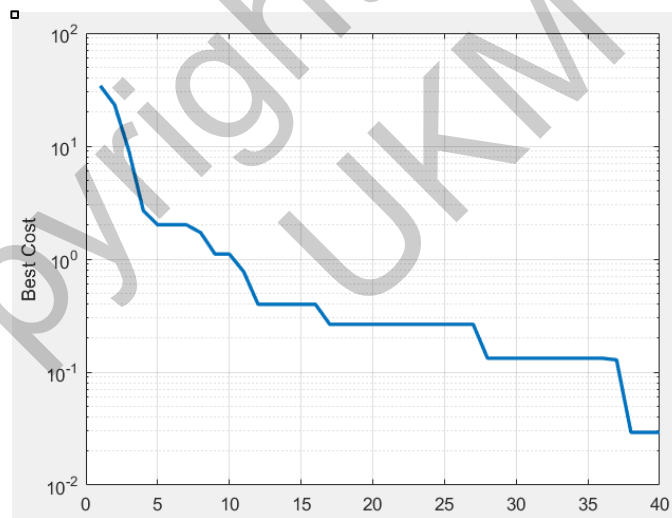
Tujuan fasa ini adalah untuk menguji algoritma PSO dan ABC. Kedua-dua algoritma diuji tanpa persekitaran terlebih dahulu untuk mendapatkan hasil. Jadual 1 menunjukkan senarai input yang diterima oleh kedua-dua algoritma manakala jadual 2 menunjukkan hasil simulasi pengujian algoritma tanpa persekitaran. Rajah 8 menunjukkan hasil bagi algoritma PSO dan rajah 9 menunjukkan hasil bagi algoritma ABC.

No.	Nama input	Nilai input
1	Bilangan pembolehkan keputusan	5
2	Bilangan lelaran maksimum	40
3	Bilangan populasi	20

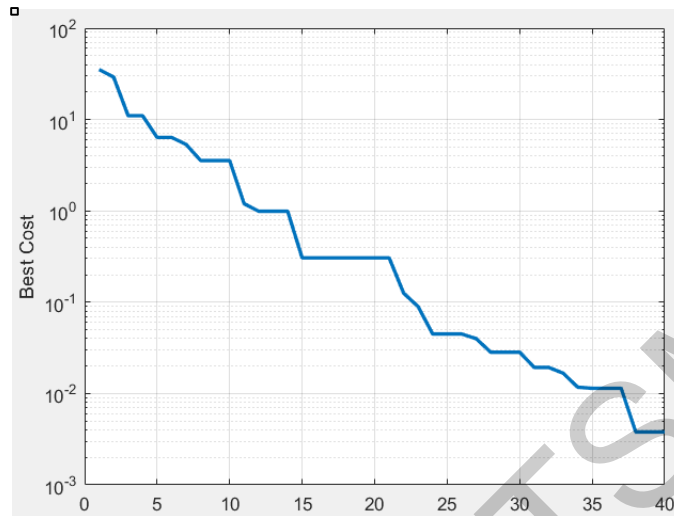
Jadual 1 Senarai input

No.	Algoritma	Kos terbaik
1	Pengoptimuman kawanan partikel	0.0293
2	Koloni lebah buatan	0.0038

Jadual 2 Hasil pengujian



Rajah 8 Hasil algoritma PSO tanpa persekitaran



Rajah 9 Hasil algoritma ABC tanpa persekitaran

Hasil menunjukkan kos terbaik bagi PSO adalah sebanyak 0.0293 manakala kos terbaik bagi ABC adalah sebanyak 0.0038 pada lelaran terakhir iaitu lelaran ke-40. Walau bagaimanapun, kedua-dua algoritma telah mencapai kos terbaik pada lelaran ke-38.

Berdasarkan rajah 8, graf kos terbaik bagi algoritma PSO menunjukkan bahawa kos yang dikemaskini adalah sangat jarang. Ini dapat dilihat pada lelaran ke-12 hingga ke-16, lelaran ke-17 hingga lelaran ke-27 dan lelaran ke-28 hingga ke-37 yang menunjukkan bahawa kos adalah malar.

Berdasarkan rajah 9 pula, graf kos terbaik bagi algoritma ABC menunjukkan bahawa kos yang dikemaskini adalah lebih kerap. Hampir setiap lelaran mempunyai kos yang telah dikemaskini. Hal ini menunjukkan cara kedua-dua algoritma tersebut untuk memperoleh kos terbaik adalah berbeza.

Dalam projek ini, kos terbaik yang minimum adalah laluan yang paling optimum. Jadi, hasil bagi simulasi tanpa persekitaran ini menunjukkan bahawa algoritma ABC adalah lebih baik berbanding algoritma PSO.

5 HASIL KAJIAN

Bahagian ini akan menerangkan serta membandingkan hasil pembangunan dan simulasi perancangan laluan bagi robot mudah alih. Algoritma akan dinilai berdasarkan 2 objektif utama bagi projek ini. Algoritma yang dapat memenuhi kedua-dua objektif tersebut adalah algoritma yang terbaik untuk perancangan laluan bagi robot mudah alih.

Simulasi ini adalah menggunakan perisian MATLAB. Terdapat dua jenis persekitaran yang dibangunkan iaitu persekitaran dengan halangan bulatan dan persekitaran dengan ruangan bilik. 3 kali simulasi akan dilakukan dalam setiap persekitaran menggunakan koordinat asal dan koordinat destinasi. Jadual 3 menunjukkan penyediaan bagi persekitaran.

No.	Jenis persekitaran	Koordinat asal	Koordinat destinasi
1a	Halangan bulatan	(0,0)	(4,6)
1b		(5,2)	(0,6)
1c		(6,4)	(0,2)
2a	Ruangan bilik	(8,8)	(12,18)
2b		(14,4)	(4,12)
2c		(8,3)	(16,14)

Jadual 3 Penyediaan persekitaran

Bagi simulasi untuk persekitaran 1 iaitu halangan bulatan, bilangan lelaran maksimum adalah 100 dan bilangan populasi adalah 50 bagi kedua-dua algoritma. Jadual 4 dan jadual 5 menunjukkan parameter bagi algoritma PSO dan algoritma ABC di persekitaran halangan bulatan.

No.	Nama parameter	Nilai parameter
1	Bilangan titik pengendalian	2
2	Bilangan lelaran maksimum	100
3	Bilangan populasi	50
4	Berat inersia	1
5	Nisbah redaman berat inersia	0.98
6	Pekali pembelajaran peribadi	1.5
7	Pekali pembelajaran global	1.5
8	Kadar pekali mutasi	0.1

Jadual 4 Parameter PSO di persekitaran halangan bulatan

No.	Nama parameter	Nilai parameter
1	Bilangan titik pengendalian	2
2	Bilangan lelaran maksimum	100
3	Bilangan populasi	50
4	Pekali pecutan	1
5	Pekali pembelajaran peribadi	1.5
6	Pekali pembelajaran global	1.5
7	Kadar pekali mutasi	0.1

Jadual 5 Parameter ABC di persekitaran halangan bulatan

Bagi simulasi untuk persekitaran 2 iaitu ruangan bilik, bilangan lelaran maksimum adalah 300 dan bilangan populasi adalah 150 bagi kedua-dua algoritma. Jadual 6 dan jadual 7 menunjukkan parameter bagi algoritma PSO dan algoritma ABC di persekitaran ruangan bilik.

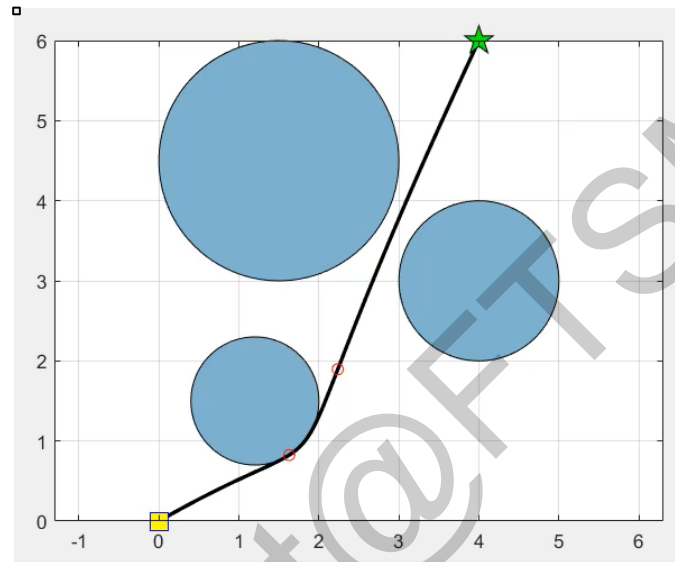
No.	Nama parameter	Nilai parameter
1	Bilangan titik pengendalian	2
2	Bilangan lelaran maksimum	300
3	Bilangan populasi	150
4	Berat inersia	1
5	Nisbah redaman berat inersia	0.98
6	Pekali pembelajaran peribadi	1.5
7	Pekali pembelajaran global	1.5
8	Kadar pekali mutasi	0.1

Jadual 6 Parameter PSO bagi persekitaran ruangan bilik

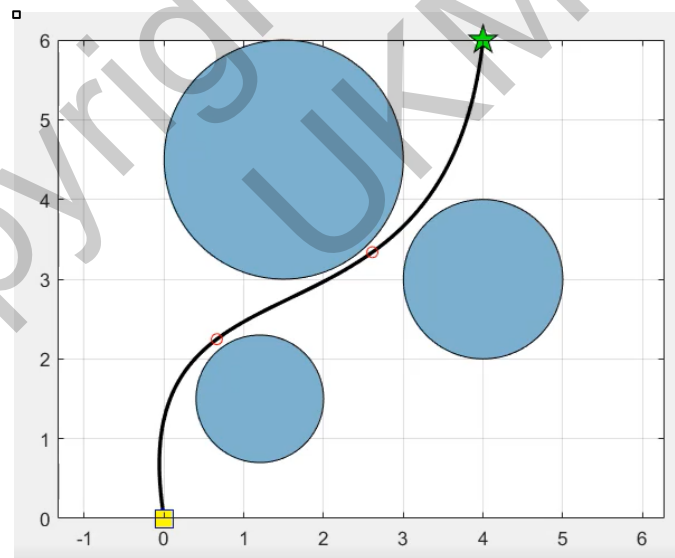
No.	Nama parameter	Nilai parameter
1	Bilangan titik pengendalian	2
2	Bilangan lelaran maksimum	300
3	Bilangan populasi	150
4	Pekali pecutan	1
5	Pekali pembelajaran peribadi	1.5
6	Pekali pembelajaran global	1.5
7	Kadar pekali mutasi	0.1

Jadual 7 Parameter ABC bagi persekitaran ruangan bilik

Berikut merupakan hasil daripada kesemua simulasi yang telah dilakukan berdasarkan penyediaan yang telah ditetapkan. Rajah 10 menunjukkan hasil algoritma PSO bagi simulasi 1a manakala rajah 11 menunjukkan hasil algoritma ABC bagi simulasi 1a.



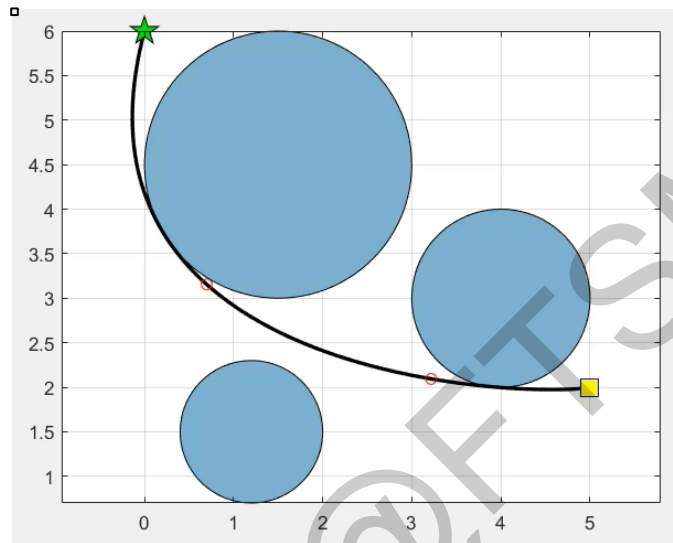
Rajah 10 Simulasi 1a PSO



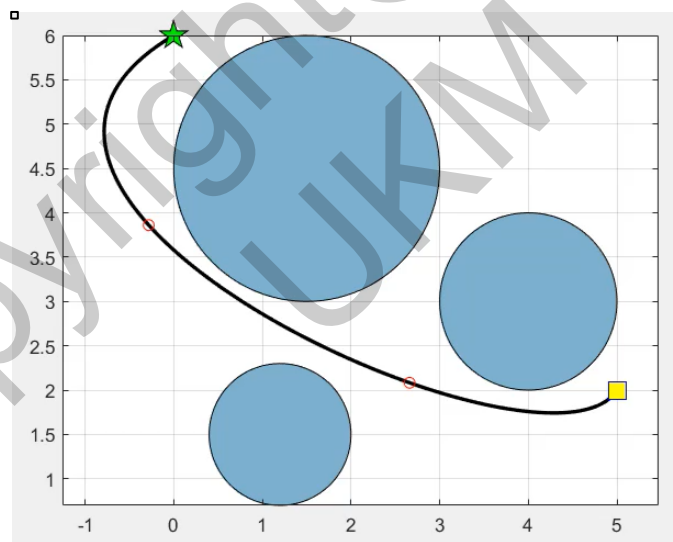
Rajah 11 Simulasi 1a ABC

Bagi simulasi 1a, terdapat pelanggaran pada PSO manakala ABC tidak mempunyai pelanggaran dan kos terbaik adalah 7.5478 bagi PSO dan 7.8197 bagi ABC. Algoritma yang lebih baik untuk simulasi 1a adalah ABC.

Rajah 12 menunjukkan hasil algoritma PSO bagi simulasi 1b manakala rajah 13 menunjukkan hasil algoritma ABC bagi simulasi 1b.



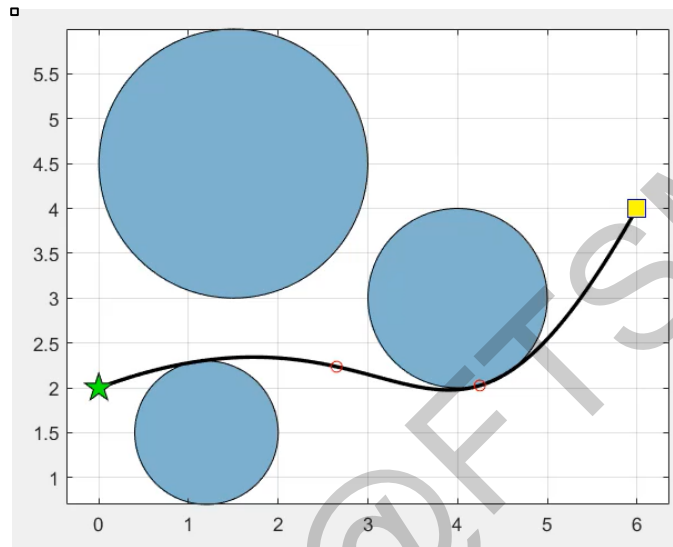
Rajah 12 Simulasi 1b PSO



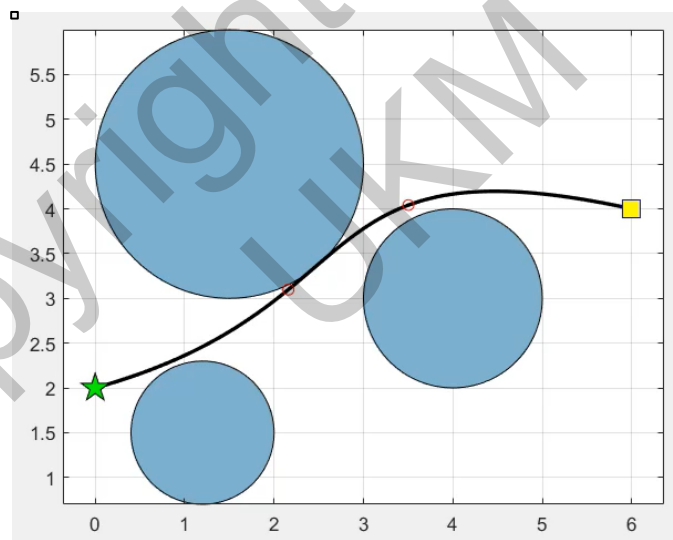
Rajah 13 Simulasi 1b ABC

Bagi simulasi 1b di halangan bulatan, terdapat perlanggaran pada PSO manakala ABC tidak mempunyai perlanggaran dan kos terbaik adalah 7.6601 bagi PSO dan 8.5231 bagi ABC. Algoritma yang lebih baik untuk simulasi 1b adalah ABC.

Rajah 14 menunjukkan hasil algoritma PSO bagi simulasi 1c manakala rajah 15 menunjukkan hasil algoritma ABC bagi simulasi 1c.



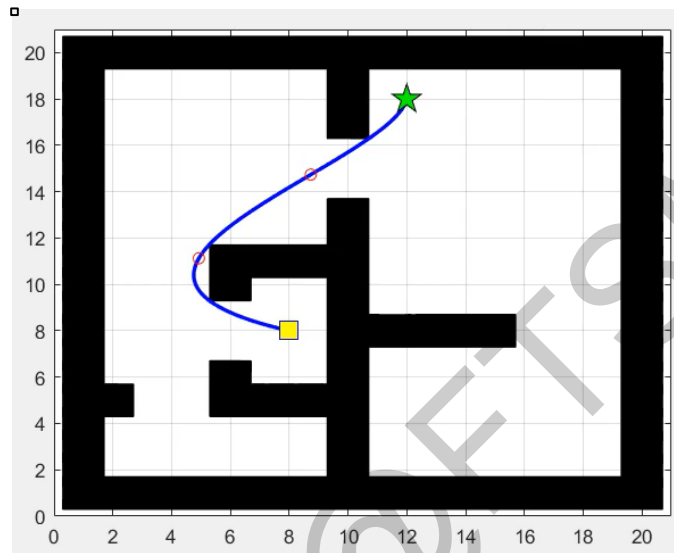
Rajah 14 Simulasi 1c PSO



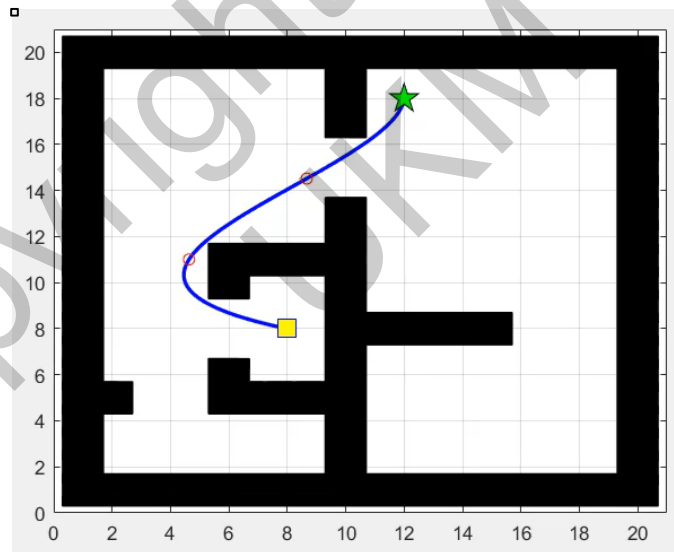
Rajah 15 Simulasi 1c ABC

Bagi simulasi 1c di halangan bulatan, terdapat perlanggaran pada kedua-dua algoritma dan kos terbaik adalah 7.0279 bagi PSO dan 6.6464 bagi ABC. Algoritma yang lebih baik untuk simulasi 1c adalah ABC kerana mempunyai kos terbaik yang minimum.

Rajah 16 menunjukkan hasil algoritma PSO bagi simulasi 2a manakala rajah 17 menunjukkan hasil algoritma ABC bagi simulasi 2a.



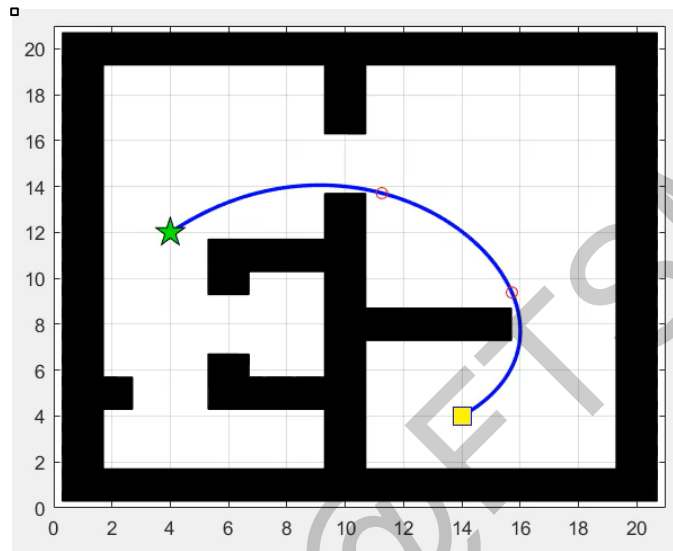
Rajah 16 Simulasi 2a PSO



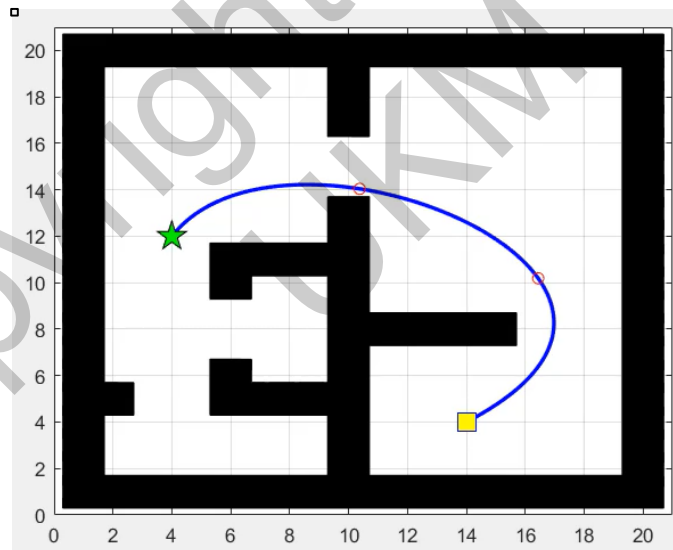
Rajah 17 Simulasi 2a ABC

Bagi simulasi 2a di ruangan bilik, terdapat pelanggaran pada PSO manakala ABC tidak mempunyai pelanggaran dan kos terbaik adalah 15.0254 bagi PSO dan 15.5209 bagi ABC. Algoritma yang lebih baik untuk simulasi 2a adalah ABC.

Rajah 18 menunjukkan hasil algoritma PSO bagi simulasi 2b manakala rajah 19 menunjukkan hasil algoritma ABC bagi simulasi 2b.



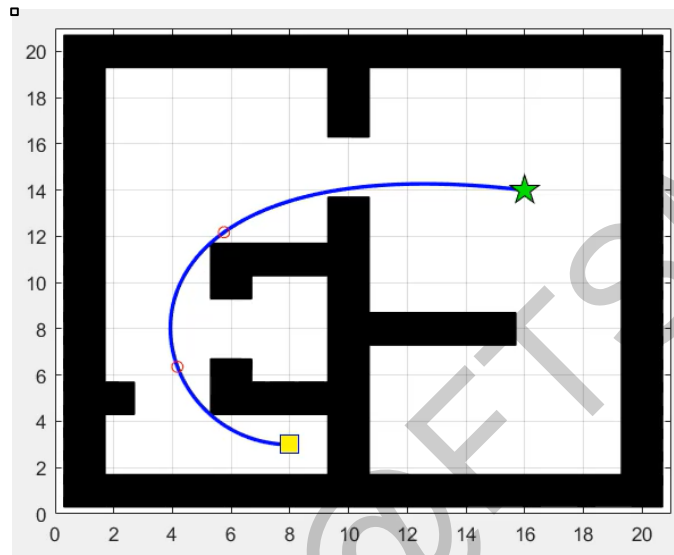
Rajah 18 Simulasi 2b PSO



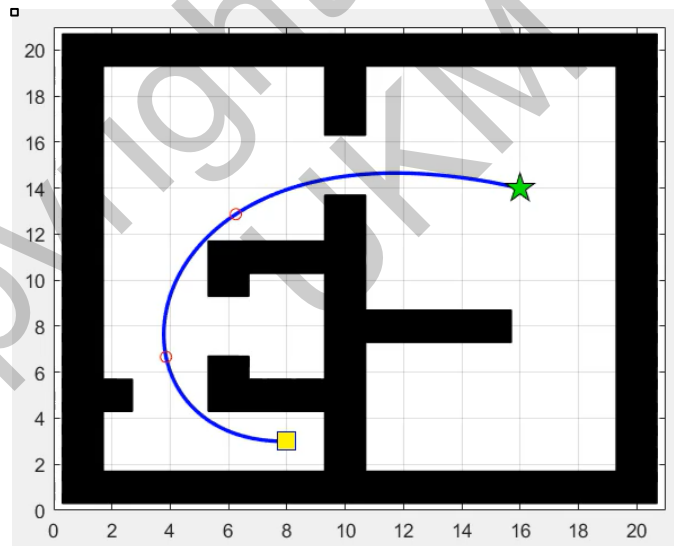
Rajah 19 Simulasi 2b ABC

Bagi simulasi 2b di ruangan bilik, kedua-dua algoritma tersebut tidak mempunyai perlanggaran dan kos terbaik adalah 20.3451 bagi PSO dan 21.9643 bagi ABC. Algoritma yang lebih baik untuk simulasi 2b adalah PSO.

Rajah 20 menunjukkan hasil algoritma PSO bagi simulasi 2c manakala rajah 21 menunjukkan hasil algoritma ABC bagi simulasi 2c.



Rajah 20 Simulasi 1c PSO



Rajah 21 Simulasi 1c ABC

Bagi simulasi 2c di ruangan bilik, terdapat pelanggaran pada PSO manakala ABC tidak mempunyai pelanggaran dan kos terbaik adalah 22.5624 bagi PSO dan 23.2517 bagi ABC. Algoritma yang lebih baik untuk simulasi 2c adalah ABC.

Jadual 8 menunjukkan hasil keseluruhan simulasi bagi algoritma PSO dan algoritma ABC di kedua-dua persekitaran.

No.	Jenis persekitaran	PSO	ABC	PSO	ABC	Algoritma yang lebih baik
		Perlanggaran		Kos terbaik		
1a	Halangan bulatan	/	X	7.5478	7.8197	ABC
1b		/	X	7.6601	8.5231	ABC
1c		/	/	7.0279	6.6464	ABC
2a	Ruang bilik	/	X	15.0254	15.5209	ABC
2b		X	X	20.3451	21.9643	PSO
2c		/	X	22.5624	23.2517	ABC

Jadual 8 Hasil keseluruhan simulasi

Berdasarkan objektif kajian yang pertama, algoritma ABC membuat perancangan laluan tanpa perlanggaran sebanyak 83% manakala PSO hanya mencatat 17%. Berdasarkan objektif kedua pula, perancangan laluan kedua-dua algoritma tersebut adalah sangat optimum kerana jarak laluan bagi robot adalah minimum. Secara keseluruhannya, algoritma ABC adalah lebih baik berbanding algoritma PSO dalam membuat perancangan laluan bagi robot mudah alih.

6 KESIMPULAN

Secara keseluruhannya, kajian perancangan laluan bagi robot mudah alih telah berjaya dibangunkan dalam tempoh masa yang diberikan dengan kejayaan pencapaian maklumat, objektif, menepati skop yang didefinisikan serta mengikuti metodologi yang dicadangkan sehingga kajian selesai. Pembangun kajian dan penyelidik dapat memperoleh maklumat dengan kemudahan yang dibekalkan iaitu maklumat tentang algoritma dan kod-kod yang berkaitan. Banyak input yang dapat diperoleh serta banyak ilmu yang dipelajari sepanjang kajian ini.

7 RUJUKAN

- M. Shabab Alam, M. Usman Rafique and M. Umer Khan, "Mobile Robot Path Planning in Static Environments using Particle Swarm Optimization". *International Journal of Computer Science and Electronics Engineering (IJCSEE)* Volume 3, Issue 3 (2015).
- Xun Li, Dandan Wu, Jingjing He, Muhammad Bashir, and Ma Liping, "An Improved Method of Particle Swarm Optimization for Path Planning of Mobile Robot". *Journal of Control Science and Engineering* Volume 2020.
- Evan Krell, Alaa Sheta, Arun Prassanth Ramaswamy Balasubramanian and Scott A. King, "Collision-Free Autonomous Robot Navigation In Unknown Environments Utilizing Pso For Path Planning". *JAISCR*, 2019, Vol. 9, No. 4.
- Divya Agarwal and Pushpendra S. Bharti, "MATLAB Simulation of Path Planning and Obstacle Avoidance Problem in Mobile Robot using SA, PSO and FA". 2020 IEEE International Conference for Innovation in Technology (INOCON).
- Vaishali R Kulkarni and Veena Desai, "ABC and PSO: A Comparative Analysis". 2016 IEEE International Conference on Computational Intelligence and Computing Research.
- Pinar Civicioglu and Erkan Besdok, "A conceptual comparison of the Cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms". *Artif Intell Rev* (2013).
- Xiaohu Shi, Yanwen Li, Haijun Li, Renchu Guan, Liupu Wang and Yanchun Liang, "An Integrated Algorithm Based on Artificial Bee Colony and Particle Swarm Optimization". 2010 Sixth International Conference on Natural Computation (ICNC 2010).
- Zhongya Wang, Min Li, Lianhang Dou, Yang Li, Qingying Zhao and Jie Li, "A Novel Multi-objective Artificial Bee Colony Algorithm for Multi-robot Path Planning". 2015 IEEE International Conference on Information and Automation.
- Mostapha Kalami Heris, "Particle Swarm Optimization in MATLAB (URL: <https://yarpiz.com/50/ypea102-particle-swarm-optimization>), Yarpiz, 2015.