Intelligent Control for a Swarm of Two Wheel Mobile Robot with Presence of External Disturbance

Mehdi J. Marie
Ministry of Industry and Minerals, Baghdad, 00964, Iraq
Email: mehdielo@gmail.com

Safiaa S.Mahdi and Esraa Y. Tarkan
Al -Nahrain University /Computer Engineering, Baghdad, 00964, Iraq
Email: { Dr_safaisoud@yahoo.com, esraayahia90@gmail.com}

Received: 30 September 2019; Accepted: 28 October 2019; Published: 08 November 2019

Abstract—This paper proposed an optimization algorithm in order to improve path maintaining of swarm of two wheel mobile robots with presence of external disturbance. The three robots forms use the leader-follower strategy, the best path for leader is determined using A* algorithm ,the other two robots follow the leader path. Two PID controller are used in each robot to control the angular and velocity torque of wheel. Each PID controller is tuned using intelligent optimization control method which are Particle swarm optimization ,random occurring distributed time delay particle swarm optimization and hybrid particle swarm optimization and genetic after that the proposed algorithm is used for tuning. The new algorithm is the contribution of this article. It is built by combine the random occurring distributed time delayed and genetic algorithm .The combination of these two algorithms takes the advantage of them by using the historical best global position of particles in random occurring distributed time delayed particle swarm optimization algorithm to update velocity of new population generated by genetic algorithm. The integral absolute error (IAE) is computed for system in each algorithm for comparison between them. The performance of intelligent control systems for controlling the three robots path is tested with presence of external disturbance in environment .Two type of external disturbance is tested, these are constant external disturbance and dynamic external disturbance. The performance of the same optimization algorithm is tested in pure environments. From the obtained result ,the new combination method is the best in both disturbance environments (constant or dynamic) and pure.

Index Terms—PSO, particle swarm optimization, GA, genetic algorithms ,IAE, integral absolute error..
The parameters of these controllers can be optimized using intelligent systems to minimize the error. There are many optimization algorithms which search for the best solution. Particle swarm optimization is one of heuristic optimization methods based on bird flocking with nature-inspired, decentralized, and self-organized characteristics. It consists of many particles which move toward the best personal local position, basically these particles randomly interact with each other to move toward the best global position with appropriate velocity. The velocity \( V \) and position \( P \) of particles are updated by the following equations:

\[
\begin{align*}
    v_i(t+1) &= \omega v_i(t) + r_1 c_1 (p_{best}(t,i) - p_i(t)) + r_2 c_2 (g_{best} - p_i(t)) & (1) \\
    p_i(t+1) &= p_i(t) + v_i(t+1) & (2)
\end{align*}
\]

where \( \omega \) is the inertia weight and \( r_1, r_2 \) are random numbers between \([0,1] \), \( c_1, c_2 \) are positive constant numbers denoting the acceleration coefficient. The first part of (1) called “inertia” and it presents the previous velocity while the second part called “cognitive” it encourages the particle to move toward the best own position. Third part known as “cooperation”, it shows the collaborative effect of the particle to find the global optimal solution. Fig. 1 presents a flowchart of general PSO.[3]

Many advances of PSO are introduced such as modification of PSO which contain a little change in origin one, hybridization of PSO which combine the PSO with the other article intelligent algorithm like GA. Fuzzy an others.

PSO with linearly decreased inertia weight (PSO_LDDID) was one of these algorithms that update the inertia weight as follows:

\[
    \alpha = \omega_{\text{max}}(\omega_{\text{max}} - \omega_{\text{min}}) \times (\text{iter} / \text{maxiter}) \quad (3)
\]

where \( \omega_{\text{max}}, \omega_{\text{min}} \) denote the maximum and minimum value of the inertia weight respectively, and \( \text{iter} \) denotes the current iteration, \( \text{maxiter} \) presents the maximum number of iteration. Large value of inertia weight good for global exploration and smaller value contribute to the local optimization.[5, 6]

Another type of PSO algorithm was proposed with a time-varying acceleration coefficients (PSO-TVAS). \( C_1 \), the cognitive acceleration coefficient is linearly decreased. \( C_2 \), the social acceleration coefficient is linearly increased which is shown as follows:

\[
\begin{align*}
    C_1 &= (c_{1f} - c_{1i}) \times (\text{maxiter} - \text{iter}) + c_{1i} \quad (4) \\
    C_2 &= (c_{2f} - c_{2i}) \times (\text{maxiter} - \text{iter}) + c_{2i} \quad (5)
\end{align*}
\]

Where \( c_{1i}, c_{2i} \) are the initial value for cognitive and social coefficient respectively and \( c_{1f}, c_{2f} \) are the final value for them.[7]

A novel RODDPSO is proposed for reducing the possibility of getting trapped in the local optima and expand the search space. In this algorithm a randomly occurring time delays that reflect the history of previous best local and global position is introduced in velocity updating model as declare below:

\[
\begin{align*}
    v_i(k + 1) &= \omega v_i(k) + r_1 c_1 (p_{i}(k) - x_i(k)) + r_2 c_2 (p_{g}(k) - x_i(k)) + m_a(\xi) \sum_{k=N}^{\infty} \alpha(\tau)(p_{i}(k - \tau) - x_i(k)) \\
    &+ m_b(\xi) \sum_{k=N}^{\infty} \alpha(\tau)(p_{g}(k - \tau) - x_i(k)) \quad (6)
\end{align*}
\]

Where \( \omega \) is the inertia weight declare in (3). \( c_1, c_2 \) are the acceleration factor calculated by (4) and (5). \( c_3, c_4 \) are the acceleration coefficient of distributed time delay time delay, \( N \) is the upper bound of time delay dimension. \( \alpha(\tau) \) represent the vector of time delay that occur randomly. \( m_a(\xi), m_b(\xi) \) are the intensity factor of the distributed time delay for evolutionary state (\( \xi \)). The evolutionary state can be one of the following convergence state denoted by \( \xi=1 \), when the particles try to fly into the global optimal region as soon as possible.

Exploitation state denoted by \( \xi=2 \), when the particles exploit the region around personal best particles.

Exploration state denoted by \( \xi=3 \), the particles explore the entire search space thoroughly.

Jumping out state denoted by \( \xi=4 \). In this state the particles are try to escape out of search space.[8]

Table 1. Show the updating strategy for distributed time delay.

<table>
<thead>
<tr>
<th>State</th>
<th>Mode</th>
<th>( m_l(\xi) )</th>
<th>( m_g(\xi) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Convergence</td>
<td>( \xi(k) = 1 )</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Exploitation</td>
<td>( \xi(k) = 2 )</td>
<td>0.01</td>
<td>0</td>
</tr>
<tr>
<td>Exploration</td>
<td>( \xi(k) = 3 )</td>
<td>0</td>
<td>0.01</td>
</tr>
<tr>
<td>Jumping-out</td>
<td>( \xi(k) = 4 )</td>
<td>0.01</td>
<td>0.01</td>
</tr>
</tbody>
</table>

Fig. 1. Flowchart of general PSO. [4]
II. PROBLEM DEFINITION

One of major problem in multi robot system is how to maintain the path for swarm of robots in turbulent environments. The effect of disturbance can be eliminated by using a specific controller. Intelligent controller one of method that can be used for this purpose. In this article, tuning PID controller with new modified optimization algorithm is introduced as a one solution for this problem.

III. MULTI AGENT SYSTEM

The multi-agent systems may be a parameter for specific processing, different parts of machine/robot or a group of robots. In a multi robot systems, many robots moving together according to specific topology and path. Moving in formation produces many benefits such as increase the efficiency and robustness of system, reducing the required cost in systems specially in configuration and adding new elements. The multi agent system in this article consist of swarm of two wheel mobile robot. In this article the three robots is built by MATLAB 2018a Simulink as shown in Fig. 2 and Fig. 3 which presents the Simulink model for single and three robots respectively while the physical parameters of this robot is selected from (Altamiro, 2004) and represented as follow 

\[
A = \begin{bmatrix}
a_1 & 0 & 0 & 1 \\
0 & 0 & 1 & a_2 \\
0 & 0 & 0 & a_3 \\
\end{bmatrix}, 
B = \begin{bmatrix}
b_1 \\
b_2 \\
\end{bmatrix}, 
n = \begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
\end{bmatrix}
\]

\[
a_1 = \frac{-2c}{(Mr^2 + 2lr)} = -2.6876 \\
(8)
\]

\[
a_2 = \frac{-2ci^2}{(l_0r^2 + 2lr^2)} = -5.1193 \\
(9)
\]

\[
b_1 = \frac{kr}{(Mr^2 + 2lr)} = 0.824147 \\
(10)
\]

\[
b_2 = \frac{-kr}{(l_0r^2 + 2lr^2)} = 2.2087 \\
(11)
\]

The robot path is determine using A* algorithm with cell decomposition method. The map of work space is converted to binary form, each obstacles represents by one and free space represent by zero. The path is computed for single robot and other robots path determine according to the leader follower equations as follow:

\[
X_2 = X_1 - b_{21} \cos (Y_{21}), 
\]

\[
y_2 = y_1 - b_{21} \sin (Y_{21}) 
(12)
\]

\[
X_3 = X_1 - b_{31} \cos (Y_{31}), 
\]

\[
y_3 = y_1 - b_{31} \sin (Y_{31}) 
(13)
\]

Where 2,3 (follower) and (leader) represented by 1, b_{21}, b_{31} are the distance between leader and follower2 and follower1 respectively, Y_{21}, Y_{31} are the angle between line projecting from the coordinates of follower2 and follower2 to the leader. PID controllers shown in Fig. 2 are tuned using PSO, HPSOGA, RODDPSO and new proposed method in order to eliminate the effect of external disturbance that may occur in system. The integral absolute error (IAE) of system is computed for each algorithm. The error in this experiments is the summation of angle and velocity error.

IV. PROPOSED METHOD

The new proposed method combine the randomly occurring distributed time delay particle swarm optimization with genetic algorithm, the historical best local and global position are used to update the velocity model. The evolutionary states are classified into four region (convergence, exploitation, exploration, jumping – out) for using them to specify the delayed intensity factor that used in velocity update. The speed and velocity updating model use a new population of particles which generated by the genetic algorithm rather than a current one. So new population is updated using early delayed information of personal and global optimum. The main goal of this proposed method is to prevent divergences of particles form the local optimum in addition to avoid the premature convergence. The parameters used in the Simulink present in table 3.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moment of inertia around C.G. (I_c)</td>
<td>0.0732 Kg.M^2</td>
</tr>
<tr>
<td>Distance between right and left wheel (l)</td>
<td>0.36m.</td>
</tr>
<tr>
<td>Viscous friction factor (c)</td>
<td>0.15833 Kg/s.</td>
</tr>
<tr>
<td>Mass (M)</td>
<td>24 Kg.</td>
</tr>
<tr>
<td>Moment of inertia of wheel (I_w)</td>
<td>0.0198 Kg.M^2</td>
</tr>
<tr>
<td>Radius of wheel (R)</td>
<td>0.057m</td>
</tr>
<tr>
<td>Driving gain factor (K)</td>
<td>1.7.</td>
</tr>
</tbody>
</table>

Fig.2. The Simulink Model of Single Robot

The dynamic model of robot represents according to the following equation:

\[
\dot{x} = Ax + Bu, \quad Y = Cx 
(7)
\]

Where:
The steps of new proposed algorithm can surmised by the following steps

1- Initialize all the parameters of RODDPSO as shown in table 3.
2- Generate the first population randomly, evaluate the fitness of all particles and define the global and local best positions.
3- Calculate the mean distance (d) between particle i and other particles, this operation repeated for all particles and it is represented by the following:

\[
d_i = \frac{1}{S-1} \sum_{j=1,j \neq 1}^{S} \sqrt{\sum_{k=1}^{D}(x_{ik} - x_{jk})^2}
\]  

(14)

Where S is the size of swarm and D is the dimension of particle

4- Use the following equation to compute the evolutionary factor \(E_i\) of each particle

\[
E_i = \frac{d_{g_i} - d_{min}}{d_{max} - d_{min}}
\]  

(15)

Where \(d_{g_i}\) is the global best mean distance particle , \(d_{min}\) and \(d_{max}\) is the minimum and maximum mean distance of \(d_i\).

5- Classify the evolutionary state \(\xi_k\) according to equal division strategy as follow

\[
\xi_k = \begin{cases} 
1, & 0.00 \leq E_i < 0.25 \\
2, & 0.25 \leq E_i < 0.50 \\
3, & 0.50 \leq E_i < 0.75 \\
4, & 0.75 \leq E_i < 1.00 
\end{cases}
\]

6- Update the inertia weight according to (3).
7- Use table1 to update the randomly distributed delay information.
8- Generate new population of particles using genetic algorithms.
9- Update velocity and speed for each particles.

### Table 3. Parameters of Simulink [9]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of swarm</td>
<td>3</td>
</tr>
<tr>
<td>Number of particles in each swarm</td>
<td>20</td>
</tr>
<tr>
<td>Acceleration factors c1,c2</td>
<td>1.1.3</td>
</tr>
<tr>
<td>Inertia weight w</td>
<td>Updating according to (3) (Dynamic)</td>
</tr>
<tr>
<td>Sample time</td>
<td>0.005</td>
</tr>
<tr>
<td>Solver</td>
<td>20</td>
</tr>
<tr>
<td>Upper timer</td>
<td>0.15</td>
</tr>
<tr>
<td>Mutation rate</td>
<td>0.01</td>
</tr>
</tbody>
</table>

The simulink model for this experiment is presented in Fig.3. From Fig.3, the general controller generate the desired path for three robots using A* algorithm. The PID parameters selected by optimisation algorithm. The robot block present the dynamic model of two wheel mobile robot. Disturbance generator add disturbance to system either constant or dynamic. Disturbance(n1) present the place where the disturbance is added.

### VI. SIMULATION RESULTS

This experiment tests the ability of maintaining the desired path for swarm of three robots with presence of external disturbance. This disturbance is supposed to occur from PID controller to the right wheel of mobile robot. Three cases of disturbance are tested: these are: constant disturbance with value of 100 and 1000, dynamic disturbance and pure environments. The test is applied with a varying number of iteration for each algorithm. PID controller is tuned with four algorithms (PSO , RODDPSO, HPSOGA, Proposed algorithm) in order to determine which of them can enhance the performance of path tracking.

Table 4 shows the total IAE for three robots with presence of constant external disturbance of value (100). Each algorithm is tested with varying number of iterations. It is clear that the minimum IAE is produced by the proposed algorithm with value 1.19248E-05 at iterations 40, the second best result produced by HPSOGA with IAE value 1.92E-05 as demonstrated in Fig. 4.

Table 5 shows the IAE for three robots with presence of constant 1000 as constant external disturbance. From table, the minimum IAE is produced by proposed algorithm with value 3.143E-05 in 40 as shown in Fig.5.

Table 6: shows the IAE for three robots with presence of dynamic disturbance in 40 iterations. Also the minimum IAE is produced by proposed algorithm with value 1.174E-05.
The new proposed algorithm proved their efficiency in pure environments by producing minimum IAE with value 2.64E-05.

The results show that’s proposed algorithm produced minimum IAE for system than other three methods with slightly decreasing of error throw higher iterations while the other method alternate in their performance throw iterations.

Table 4. Error for three robots with 100 constant disturbance

<table>
<thead>
<tr>
<th>N.</th>
<th>PSO</th>
<th>RODDPSO</th>
<th>HPSOGA</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>4.84E-05</td>
<td>4.20374E-05</td>
<td>4.69E-05</td>
<td>4.03639E-05</td>
</tr>
<tr>
<td>20</td>
<td>4.99E-05</td>
<td>7.925E-05</td>
<td>6.45E-05</td>
<td>2.20777E-05</td>
</tr>
<tr>
<td>30</td>
<td>3.64E-05</td>
<td>2.94769E-05</td>
<td>1.92E-05</td>
<td>3.40148E-05</td>
</tr>
<tr>
<td>40</td>
<td>3.33E-05</td>
<td>2.98984E-05</td>
<td>2.31E-05</td>
<td>1.19248E-05</td>
</tr>
</tbody>
</table>

Table 5. IAE for three robots with 1000 constant disturbance

<table>
<thead>
<tr>
<th>N.</th>
<th>PSO</th>
<th>RODDPSO</th>
<th>HPSOGA</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.0001213</td>
<td>0.000157</td>
<td>0.0002328</td>
<td>0.0001156</td>
</tr>
<tr>
<td>20</td>
<td>7.988E-05</td>
<td>0.0001031</td>
<td>0.0001843</td>
<td>4.833E-05</td>
</tr>
<tr>
<td>30</td>
<td>9.391E-05</td>
<td>6.139E-05</td>
<td>0.0001401</td>
<td>6.95E-05</td>
</tr>
<tr>
<td>40</td>
<td>8.083E-05</td>
<td>5.464E-05</td>
<td>0.0001212</td>
<td>3.143E-05</td>
</tr>
</tbody>
</table>

Table 6. IAE for three robots with Dynamic Disturbance for 40 iterations

<table>
<thead>
<tr>
<th>Error</th>
<th>PSO</th>
<th>RODDPSO</th>
<th>HPSOGA</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Robot1</td>
<td>2.907E-05</td>
<td>2.93E-05</td>
<td>1.235E-05</td>
<td>6.403E-06</td>
</tr>
<tr>
<td>Robot2</td>
<td>6.905E-07</td>
<td>1.20E-05</td>
<td>1.07E-05</td>
<td>1.276E-06</td>
</tr>
<tr>
<td>Robot3</td>
<td>2.221E-05</td>
<td>8.51E-06</td>
<td>1.59E-05</td>
<td>4.06E-06</td>
</tr>
<tr>
<td>Total</td>
<td>5.197E-05</td>
<td>4.978E-05</td>
<td>3.893E-05</td>
<td>1.174E-05</td>
</tr>
</tbody>
</table>

Table 7. IAE for three robots in free disturbance environments

<table>
<thead>
<tr>
<th>algorithm</th>
<th>Itr.</th>
<th>Robot1 Error</th>
<th>Robot2 Error</th>
<th>Robot3 Error</th>
<th>Total error</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSO</td>
<td>10</td>
<td>3.46E-05</td>
<td>3.46E-05</td>
<td>1.77E-05</td>
<td>8.70E-05</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>4.69E-05</td>
<td>1.05E-05</td>
<td>9.25E-06</td>
<td>6.66E-05</td>
</tr>
<tr>
<td>RODDPSO</td>
<td>10</td>
<td>1.03E-05</td>
<td>2.52E-06</td>
<td>4.85E-05</td>
<td>6.13E-05</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>3.73E-05</td>
<td>1.17E-06</td>
<td>1.95E-05</td>
<td>5.80E-05</td>
</tr>
<tr>
<td>HPSOGA</td>
<td>10</td>
<td>7.33E-06</td>
<td>5.07E-06</td>
<td>2.82E-05</td>
<td>4.06E-05</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>8.10E-06</td>
<td>7.76E-06</td>
<td>1.93E-05</td>
<td>3.52E-05</td>
</tr>
<tr>
<td>Proposed</td>
<td>10</td>
<td>9.30E-06</td>
<td>1.02E-05</td>
<td>7.56E-06</td>
<td>2.71E-05</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>6.33E-06</td>
<td>1.60E-06</td>
<td>1.85E-05</td>
<td>2.64E-05</td>
</tr>
</tbody>
</table>

VII. CONCLUSIONS

This articles produces a new proposed optimization algorithms for tuning the PID controller of swarm of two wheel mobile robot. The PID controller is used in each robot to control the velocity and angular torque of each wheel .The external disturbance is added to the signal form PID controller to the right wheel .The results proven that system produces minimum IAE when the PID controller is tuned by the new proposed method. The proposed method uses the historical information of local and global optimum to update the velocity and speed of new population. It add the distributed time delay to velocity equation of PSO with a particles generated by GA .The algorithm was tested in three cases which are constant external disturbance, dynamic external disturbance and free environments , the results show that proposed algorithm is the best for keeping the desired path of three robots as possible by producing the minimum IAE. In future work the new algorithm can be used for tuning the non-linear PID controller , also it can be combine with fuzzy and other intelligent system.
REFERENCES

[1] Andres F. Rivera, “Formation Control And Robustness Analysis Of Time-Delayed Agents “, MSC thesis, California State University, Long Beach, Department of Mechanical and Aerospace Engineering, 2019


Authors’ Profiles

Mehdi J. Marie, is a head of engineering department at Ministry of Industry and Minerals, he also lecturer at Al-Nahrain University. In teaching, he has been focusing on soft computing. His PhD from computer engineering.

Safaa S. Mahdi: is lecturer at Al-Nahrain University for the following field: Advance microprocessor, advance digital signal, advance image processing. His bachelor in electrical engineering at 1980 from the Military Engineering College, Iraq. His MSc in digital system at 1984 from Canfield University, united kingdom. His PhD in Digital Systems, 1987, from Canfield University, united kingdom.

Esraa. Y. Tarkan: is a computer engineering at ministry of higher education and scientific research. Her bachelor in computer engineering at 2012 from Al-Mustansria University, Iraq. Now she is MSc student at computer engineering, Al-Nahrain University, Iraq. Here interest in computer controller and computer network.

How to cite this paper: Mehdi J. Marie, Safaa S. Mahdi, Esraa Y. Tarkan, "Intelligent Control for a Swarm of Two Wheel Mobile Robot with Presence of External Disturbance", International Journal of Modern Education and Computer Science (IJMECS), Vol.11, No.11, pp. 7-12, 2019. DOI: 10.5815/ijmecs.2019.11.02