



---

## Grey Wolf Optimizer-Based Approaches to Path Planning and Fuzzy Logic-based Tracking Control for Mobile Robots

R.-E. Precup, E.-I. Voisan, E. M. Petriu, M. L. Tomescu,  
R.-C. David, A.-I. Szedlak-Stinean, R.-C. Roman

**Radu-Emil Precup, Emil-Ioan Voisan**

Politehnica University of Timisoara  
Bd. V. Parvan 2, 300223 Timisoara, Romania  
[radu.precup@aut.upt.ro](mailto:radu.precup@aut.upt.ro), [emil.voisan@upt.ro](mailto:emil.voisan@upt.ro)

**Emil M. Petriu**

University of Ottawa  
800 King Edward, Ottawa, Ontario, K1N 6N5 Canada  
[petriu@uottawa.ca](mailto:petriu@uottawa.ca)

**Marius L. Tomescu\***

Aurel Vlaicu University of Arad  
Str. Elena Dragoi 2, 310330 Arad, Romania  
\*Corresponding author: [tom\\_uav@yahoo.com](mailto:tom_uav@yahoo.com)

**Radu-Codrut David, Alexandra-Iulia Szedlak-Stinean, Raul-Cristian Roman**

Politehnica University of Timisoara  
Bd. V. Parvan 2, 300223 Timisoara, Romania  
[davidradu@gmail.com](mailto:davidradu@gmail.com), [alexandra-iulia.stinean@aut.upt.ro](mailto:alexandra-iulia.stinean@aut.upt.ro), [raul.roman@aut.upt.ro](mailto:raul.roman@aut.upt.ro)

### Abstract

This paper proposes two applications of Grey Wolf Optimizer (GWO) algorithms to a path planning (PaPl) problem and a Proportional-Integral (PI)-fuzzy controller tuning problem. Both optimization problems solved by GWO algorithms are explained in detail. An off-line GWO-based PaPl approach for Nonholonomic Wheeled Mobile Robots (NWMRs) in static environments is proposed. Once the PaPl problem is solved resulting in the reference trajectory of the robots, the paper also suggests a GWO-based approach to tune cost-effective PI-fuzzy controllers in tracking control problem for NWMRs. The experimental results are demonstrated through simple multi-agent settings conducted on the nRobotic platform developed at the Politehnica University of Timisoara, Romania, and they prove both the effectiveness of the two GWO-based approaches and major performance improvement.

**Keywords:** Grey Wolf Optimizer, Proportional-Integral-fuzzy control, path planning, tracking control.

# 1 Introduction

High performance controllers are currently developed for mobile robots in order to cope with the three main navigation control problems [25], [24], [17], [20], i.e. path planning (or following), point stabilization and tracking control (or tracking a reference trajectory, further divided in local and global tracking problems [24]). The control of Nonholonomic Wheeled Mobile Robots (NWMRs) has received much research interest during the past two decades because of the effects of nonholonomic constraints on the feasible control signals of this class of nonsmooth or nonholonomic systems. Representative approaches to tracking control are backstepping [75], adaptive [49], [23], fuzzy [10], periodic [9] and neural networks [30] control. Multi-robot Path planning (PaPl) problems are generally solved by centralized and decentralized algorithms. While decentralized algorithms generate independently (separately) collision-free paths for each robot avoiding possible inter-robot collisions, centralized algorithms consider each robot as a subsystem, thus global optimization is enabled.

Nature-inspired optimization algorithms are inserted in various versions of centralized algorithms to solve PaPl optimization problems. Recent surveys on optimal motion PaPl are conducted in [36] and [72]. Popular nature-inspired algorithms included in PaPl approaches are directly related to the evolution of these algorithms, and they include Particle Swarm Optimization (PSO) [63], [35], [37], genetic algorithms [31], [29], [28], Gravitational Search Algorithms (GSAs) [62], [63], [11], simulated annealing [35], chemical optimization [38], firefly optimization [22], Charged System Search (CSS) [56], [57], Ant Colony Optimization algorithms [74], [33], hybrid algorithms as, for example, PSO-GSA [12], neural networks and fuzzy logic [36]. Some fresh results in this field are reported in the recent papers [34], [43], [44] and [2].

The Grey Wolf Optimizer (GWO) algorithm [40] mimics grey wolf behaviour towards finding the prey, which actually models the solution to the optimization problem (OP). The advantage of all versions of GWO algorithms is the reduced number of search parameters, which is reflected in various applications including GWO-based optimal tuning of Proportional-Integral (PI)-fuzzy controllers [53], [54], [55], [13] and Proportional-Integral-Derivative (PID)-fuzzy controllers [27], [46], [70]. GWO algorithms are also applied to mobile robot PaPl, and recent results concern integration of safe boundary algorithm [65], inclusion of chaotic features in GWO [41], and combination of artificial intelligence techniques and probability theory [73], [50]. In addition, recent GWO-based PaPl approaches for underwater autonomous vehicles are reported in [64], [14], [19].

This paper is built upon our recent papers on PaPl, tracking control [59], [62], [63], [56], [57], and GWO-based optimal tuning of PI-fuzzy controllers [53], [54], [55], [13], and suggests two new contributions, namely two applications of GWO to PaPl problem and PI-fuzzy controller tuning problem. Both OPs solved by GWO algorithms will be explained in detail. No theoretical contribution is given as it is highlighted in the papers specified in this paragraph.

These new contributions have the advantages (a) to (d) over the state-of-the-art pointed out in this section: (a) the PaPl approach can be easily generalized to aim the use of other nature-inspired optimization algorithms, (b) cost-effective tracking fuzzy controller designs and implementations characterized by simple controller structures and tuning are offered, (c) both approaches benefit from the reduced number of search parameters specific to GWO, (d) real-time experimental testing of PaPl and tracking approaches conducted on the nRobotic platform developed at the Politehnica University of Timisoara, Romania, (e) fuzzy logic is involved in both GWO-based approaches and applications in terms of the current trends in the heuristic design of fuzzy inference systems [47]. The advantage (a) is also the reason of choosing the PaPl as the application of GWO algorithm.

The results are demonstrated through a multi-agent setting. Comparisons with PSO-, GSA- and CSS-based approaches are included. The experiments show that the proposed approaches exhibit performance improvement. The overall results prove that GWO invoked in this paper is a good candidate for the given PaPl problem.

The rest of the paper treats the following topics: the PaPl problem formulation and the off-line GWO-based optimal PaPl approach are presented in the next section. The GWO-based approach to tracking fuzzy control is described in Section 3. The two new approaches are tested and validated in Section 4 by a set of experimental results but also comparison with similar methods. The conclusions are outlined in Section 5.

## 2 Optimal path planning problem and gwo-based path planning approach

As considered in [57], let  $\{R_k|k = 1...p\}$  be a set of  $p$  mobile robots in a known static environment with obstacles included. This paper adopts the same number of  $N$  agents that generates the paths for each robots  $R_1...R_n$  in terms of the individual agent populations  $P_{R_1}...P_{R_p}$ . Robot's solution space is a two-dimensional search space, and agents are placed in robot's position. Using the notation  $\mathbf{X}_{i,P_{R_k}}(t)$  for the robot vector position and also  $i^{th}$  agent's current vector position,  $i = 1...N$  in the population  $P_{R_k}$ ,  $k = 1...p$ , in the search space at the time moment (iteration)  $t$ ,  $t \in \mathbf{N}$

$$\mathbf{X}_{i,P_{R_k}}(t) = (x_{i,P_{R_k}}(t), y_{i,P_{R_k}}(t)) \in \mathbf{R}^2, \quad (1)$$

where  $\mathbf{N}$  is the set of natural numbers,  $\mathbf{R}$  is the set of real numbers,  $x_{i,P_{R_k}}(t)$  is agent's position on the  $x$  axis and  $y_{i,P_{R_k}}(t)$  is agent's position on the  $y$  axis, the paths (trajectories) are generated iteratively by means of path point sets  $T_{i,P_{R_k},t}$  updated by adding the current agent position

$$T_{i,P_{R_k},t} = \{\mathbf{X}_{i,P_{R_k}}(0), \mathbf{X}_{i,P_{R_k}}(1), \dots, \mathbf{X}_{i,P_{R_k}}(t)\}, \quad (2)$$

$i = 1...N$ ,  $k = 1...p$ , with  $\mathbf{X}_{i,P_{R_k}}(0) = (x_{i,P_{R_k}}(0), y_{i,P_{R_k}}(0))$  the initial position of  $i^{th}$  agent in  $P_{R_k}$  as the initial time moment is 0.

As shown in [59], [62], [63], [56], [57], the path of each agent (i.e., robot) is built by connecting with lines all successive points of the sets  $T_{i,P_{R_k},t_{max}(i,k)}$  where  $t_{max}(i,k)$  is the maximum number of time moments (or iterations) allowed for  $i^{th}$  agent in  $P_{R_k}$ . All agents' paths that are not collision-free ( $c - f$ ), namely the path point sets  $T_{i,P_{R_k},t_{max}(i,k)}$  for which  $i \in S_{c-f,P_{R_k}} \subset \{1, 2, \dots, N\}$ , are dropped out, where  $S_{c-f,P_{R_k}}$  is the set of agents which lead to  $c - f$  paths for  $P_{R_k}$ . The sets  $T_{i,P_{R_k},t_{max}(i,k)}$  actually correspond to separate  $c - f$  paths for each robot  $R_k$ , i.e., to feasible paths. The total length of each path is

$$D_{i,k} = \sum_{t=1}^{t_{max}(i,k)} \|\mathbf{X}_{i,P_{R_k}}(t) - \mathbf{X}_{i,P_{R_k}}(t-1)\|_2, i \in S_{c-f,P_{R_k}}, \quad (3)$$

where the Euclidean norm is used.

This paper considers four PaPl objectives [59], [62], [63], [56], [57], which are inserted in separate objective functions (o.f.s), next grouped in a weighted sum, and the PaPl problem is defined as a set of OPs that minimize these o.f.s. The OPs that produce the indices  $\hat{i}_k$  of agents corresponding to the minimum length feasible paths are [63], [57].

$$\hat{i}_k = \arg \min_{i \in S_{c-f,P_{R_k}}} D_{i,k}, \quad (4)$$

and the optimal paths are  $T_{\hat{i}_k,t_{max}(\hat{i}_k,k)}$ . The elements of  $T_{i,P_{R_k},t}$  specified in (2) are solutions to the OPs assigned to each robot  $R_k$ ,  $k = 1...p$ :

$$\bar{\mathbf{X}}_{i,P_{R_k}}(t) = \arg \min_{\mathbf{X}_{i,P_{R_k}}(t)} f_{i,P_{R_k}}(\mathbf{X}_{i,P_{R_k}}(t)), t = 1...t_{max}(i,k), \quad (5)$$

$i = 1...N$ ,  $k = 1...p$ , the first element in  $T_{i,P_{R_k},t}$ , i.e.  $\mathbf{X}_{i,P_{R_k}}(0)$ , is known:

$$\bar{\mathbf{X}}_{i,P_{R_k}}(0) = \mathbf{X}_{i,P_{R_k}}(0), i = 1...N, k = 1...p, \quad (6)$$

and the expressions of o.f.s are

$$\begin{aligned} f_{i,P_{R_k}}(\mathbf{X}_{i,P_{R_k}}(t)) &= \|\mathbf{X}_f - \mathbf{X}_{i,P_{R_k}}(t)\|_2 + \lambda_1 / \left[ \sum_{j=1, j \neq k}^p \|\mathbf{X}_{best,P_{R_j}}(t) - \mathbf{X}_{i,P_{R_k}}(t)\|_2 \right] \\ &+ \lambda_2 / \left[ \sum_{j=1, j \neq k}^p |x_{best,P_{R_j}}(t) - x_{i,P_{R_k}}(t)| \right] + \lambda_3 / \left[ \sum_{j=1, j \neq k}^p |y_{best,P_{R_j}}(t) - y_{i,P_{R_k}}(t)| \right], \end{aligned} \quad (7)$$

$i = 1...N, k = 1...p.$

The vector  $\mathbf{X}_f = (x_f, y_f) \in \mathbf{R}^2$  in (7) is the final (or target) point, it is the same for all agents. The vector  $\mathbf{X}_{best, P_{R_j}}(t) = (x_{best, P_{R_j}}(t), y_{best, P_{R_j}}(t)) \in \mathbf{R}^2$  is the position of best agent (as far as the distance between the current point and the target point is concerned) in the population  $P_{R_j}$  at time moment  $t$ . The weighting parameters in (7) are  $\lambda_1 > 0, \lambda_2 > 0$  and  $\lambda_3 > 0$ .

The first term in (7) aims the minimization of the (Euclidian) agent-target position distances at a specific time (iteration)  $t$ . The second term in  $f_{i, P_{R_k}}(\mathbf{X}_{i, P_{R_k}}(t))$  aims the maximization of the distance between the agents of a population and the best agents from all other populations (for collision avoidance). The third and fourth terms in (7) aim the maximization of the distance between the robot paths on the  $x$  and  $y$  axes in order to model specific applications where one axis could be more important than the other one.

Essentially, our off-line GWO-based optimal PaPl approach consists of the steps, PaPl1, PaPl2 and PaPl3:

*Step PaPl1.* GWO algorithms are applied to solve the OPs defined in (5), with the o.f.s defined in (7) and the solutions  $\bar{\mathbf{X}}_{i, P_{R_k}}(t)$ . *Step PaPl2.* The solutions  $\bar{\mathbf{X}}_{i, P_{R_k}}(t)$  and the vector in (6) are merged to build the elements of  $T_{i, P_{R_k}, t}$  in (2), with the notation  $\mathbf{X}_{i, P_{R_k}}(t)$  for  $\bar{\mathbf{X}}_{i, P_{R_k}}(t)$ . *Step PaPl3.* The OPs defined in (4) with the o.f.s defined in (3) are solved to get the optimal paths  $T_{i, t_{\max}(\hat{i}_k, k)}$ . A simple deterministic direct search algorithm is used in this regard.

As considered in [63], [57], an additional penalty term  $\alpha_{PaPl} > 0$  is added to the right-hand term in (7) but only in certain situations in step PaPl1. Therefore, such situations will be avoided as they lead to collisions when  $i^{th}$  agent's position at iteration  $t$  is inside one of the obstacles or the line between the positions at iterations  $t$  and  $t - 1$  intersects an obstacle.

GWO algorithms, which are involved in step PPA1, consist of the steps GWOA1 to GWOA8 specific to the initial formulations of the GWO algorithm given in [53], [54], [55], [13], where a brief introduction of GWO algorithm is also provided before applying the algorithm. The presentation of the steps that are summarized below has the advantage of being supported by transparent and easily understandable vector operations.

*Step GWOA1.* The random grey wolf population, represented by  $\mathbf{N}$  agents' positions in the  $q$ -dimensional search space, is initially generated. The iteration index and the maximum number of iterations are initialized and set to  $t = 0$  and  $t_{max}$ , respectively. Using the notation  $\mathbf{X}_i(t)$  for the position vector of  $i^{th}$  agent (grey wolf)

$$\mathbf{X}_i(t) = [x_i^1(t) \quad \dots \quad x_i^f(t) \quad \dots \quad x_i^q(t)]^T \in \mathbf{R}^q, i = 1 \dots N, \tag{8}$$

where  $x_i^f(t)$  is the position of  $i^{th}$  agent in  $f^{th}$  dimension,  $f = 1 \dots q$ , and  $t = 1 \dots t_{max}$ , GWO algorithm is mapped onto the OPs defined in (5) in terms of

$$\mathbf{X}_i(t) = \mathbf{X}_{i, P_{R_k}}(t), q = 2, t_{\max} = t_{\max}(i, k), J = f_{i, P_{R_k}}, \tag{9}$$

with  $k = 1 \dots p$ , and the index  $k$  is omitted as follows for the sake of simplicity.

*Step GWOA2.* The performance of each member of agent population is evaluated by conducting simulations and/or experiments on the set of  $p$  mobile robots using the nRobotic platform. The top three agents constrain the other agents (i.e., the so-called omega ( $\omega_{GWO}$ ) agents) to update their positions according to theirs. The best three agent position vectors (solutions) computed at each iteration (also called the alpha ( $\alpha_{GWO}$ ), beta ( $\beta_{GWO}$ ), and delta ( $\delta_{GWO}$ ) solutions) are ([55]:

$$\mathbf{X}^l(t) = [x^{l1}(t) \quad \dots \quad x^{lf}(t) \quad \dots \quad x^{lq}(t)]^T, l \in \{\alpha_{GWO}, \beta_{GWO}, \delta_{GWO}\}, \tag{10}$$

and the three vector solutions  $\mathbf{X}^{\alpha_{GWO}}(t)$ ,  $\mathbf{X}^{\beta_{GWO}}(t)$  and  $\mathbf{X}^{\delta_{GWO}}(t)$  are calculated in terms of the selection process described as follows:

$$\begin{aligned} J(\mathbf{X}^{\alpha_{GWO}}(t)) &= \min_{i=1 \dots N} \{J(\mathbf{X}_i(t)) | \mathbf{X}_i(t) \in D_{\mathbf{X}}\}, \\ J(\mathbf{X}^{\beta_{GWO}}(t)) &= \min_{i=1 \dots N} \{J(\mathbf{X}_i(t)) | \mathbf{X}_i(t) \in D_{\mathbf{X}} \setminus \{\mathbf{X}^{\alpha_{GWO}}(t)\}\}, \\ J(\mathbf{X}^{\delta_{GWO}}(t)) &= \min_{i=1 \dots N} \{J(\mathbf{X}_i(t)) | \mathbf{X}_i(t) \in D_{\mathbf{X}} \setminus \{\mathbf{X}^{\alpha_{GWO}}(t), \mathbf{X}^{\beta_{GWO}}(t)\}\}, \end{aligned} \tag{11}$$

where  $D_{\mathbf{X}}$  is the feasible domain of  $\mathbf{X}_i(t)$ . These vector solutions fulfil the condition

$$J(\mathbf{X}^{\alpha_{GWO}}(t)) < J(\mathbf{X}^{\beta_{GWO}}(t)) < J(\mathbf{X}^{\delta_{GWO}}(t)). \tag{12}$$

*Step GWOA3.* The first three best solutions obtained so far, viz.  $\mathbf{X}^{\alpha_{GWO}}(t)$ ,  $\mathbf{X}^{\beta_{GWO}}(t)$  and  $\mathbf{X}^{\delta_{GWO}}(t)$ , are identified and organized accordingly using (11).

*Step GWOA4.* The set of search coefficients is developed and computed in terms of

$$\begin{aligned} a_l^f(t) &= a^f(t)(2r_{1l}^f - 1), \\ c_l^f(t) &= 2r_{2l}^f, l \in \{\alpha_{GWO}, \beta_{GWO}, \delta_{GWO}\}, \end{aligned} \tag{13}$$

where  $r_{1l}^f$  and  $r_{2l}^f$  are uniformly distributed random numbers within  $0 \leq r_{1l}^f \leq 1$ ,  $0 \leq r_{2l}^f \leq 1$ ,  $f = 1 \dots q$ , and the values of  $a^f(t)$  are linearly decreased from 2 to 0:

$$a^f(t) = 2[1 - (t - 1)/(t_{\max} - 1)], f = 1 \dots q. \tag{14}$$

*Step GWOA5.* The agents are moved to the new positions expressed as the vectors  $X_i(t + 1)$  by first calculating the approximate distances between the current solution and the alpha, beta, and delta ones

$$d_l^{if}(t) = |c_l^f(t)x^{lf}(t) - x_i^f(t)|, i = 1 \dots N, l \in \{\alpha_{GWO}, \beta_{GWO}, \delta_{GWO}\}, \tag{15}$$

the components of the updated alpha, beta, and delta ones are next calculated as

$$x^{lf}(t + 1) = x^{lf}(t) - a_l^f(t)d_l^{if}(t), f = 1 \dots q, i = 1 \dots N, l \in \{\alpha_{GWO}, \beta_{GWO}, \delta_{GWO}\}. \tag{16}$$

As a result, the updated expressions of agents' positions are the arithmetic means of updated alpha, beta, and delta agent positions (that belong to these vector solutions)

$$x_i^f(t + 1) = \frac{x^{\alpha_{GWO}f}(t + 1) + x^{\beta_{GWO}f}(t + 1) + x^{\delta_{GWO}f}(t + 1)}{3}, f = 1 \dots q, i = 1 \dots N. \tag{17}$$

The vector equivalent of (17) is

$$\mathbf{X}_i(t + 1) = \frac{\mathbf{X}^{\alpha_{GWO}}(t + 1) + \mathbf{X}^{\beta_{GWO}}(t + 1) + \mathbf{X}^{\delta_{GWO}}(t + 1)}{3}, i = 1 \dots N. \tag{18}$$

Equations (17) and (18) are important as they reduce the effects of the random parameters, which are always a problem in the search process and presentation of results specific to metaheuristic algorithms.

*Step GWOA6.* The updated vector solution obtained so far,  $\mathbf{X}_i(\mathbf{t} + 1)$ , is validated if  $\mathbf{X}_i(\mathbf{t} + 1) \in D_{\mathbf{X}}$ .

*Step GWOA7.* The iteration index  $t$  is incremented and the algorithm continues with step 2 until  $t_{\max}$  is reached.

*Step GWOA8.* GWO algorithm is stopped and the final solution obtained at each iteration consists of the following set of solutions to the OPs defined in (5):

$$\{\bar{\mathbf{X}}_{i,PR_k}(1), \dots, \bar{\mathbf{X}}_{i,PR_k}(t), \dots, \bar{\mathbf{X}}_{i,PR_k}(t_{\max})\} = \{\mathbf{X}_i(0), \dots, \mathbf{X}_i(t), \dots, \mathbf{X}_i(t_{\max})\}. \tag{19}$$

### 3 GWO-based approach to tracking fuzzy control

This section addresses the tracking control by the design and tuning of linear and next fuzzy controllers. Extending the dynamic model presented in [71], the simplified dynamic model of NWMRs with two degrees of freedom is obtained and expressed as [59].

$$\begin{aligned} \dot{x} &= v \cos \theta, \\ \dot{y} &= v \sin \theta, \\ \dot{v} &= a_v, \\ \dot{\theta} &= \omega, \\ T_{\Sigma 1} \dot{a}_v + a_v &= k_{P1}(u_1 + d_1), \\ T_{\Sigma 2} \dot{\omega} + \omega &= k_{P2}(u_2 + d_2), \end{aligned} \tag{20}$$

where (Figure 1):  $(x, y)$  - the coordinates of the centre of rear axis of NWMR,  $v$  and  $a_v$  - the forward velocity and acceleration, respectively,  $\theta$  - the angle between the heading direction and the  $x$  axis,  $\omega$  - the angular velocity;  $u_1$  and  $u_2$  - the control signals,  $d_1$  and  $d_2$  - the load-type disturbance inputs,  $k_{P1}$  and  $k_{P2}$  - the process gains,  $T_{\Sigma 1}$  and  $T_{\Sigma 2}$  - the small time constants or the time constants equivalent to the cumulative effects of actuators, measuring instrumentation, control equipment dynamics and parasitic time constants.

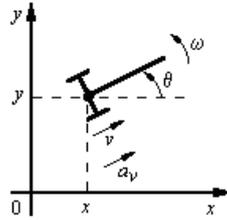


Figure 1: Mechanical variables of NWMRs with two degrees of freedom.

The structure of the simplified dynamic model given in (20), viewed as a controlled process, is also illustrated by the block diagram presented in Figure 2. Figure 2 outlines the kinematic subsystem (KS) and the dynamic system with the transfer functions of the subsystems  $H_{P1}(s)$  and  $H_{P2}(s)$  of NWMR as controlled process, defined in zero initial conditions

$$H_{P\psi}(s) = k_{P\psi} / [s(1 + T_{\Sigma\psi}s)], \psi \in \{1, 2\}. \tag{21}$$

The control system structure employed in tracking control of NWMRs with two degrees of freedom is presented in Figure 3, where: C-1 and C-2 - the forward velocity controller and angle controller, respectively,  $v_\gamma$  and  $\theta_\gamma$  - the reference inputs of the two control loops, RF-1 and RF-2 - the reference input filters;  $\tilde{v}_r$  and  $\tilde{\theta}_r$  - the filtered reference inputs of the two control loops;  $e_1 = \tilde{v}_r - v$  and  $e_2 = \tilde{\theta}_r - \theta$  - the control errors,  $x_r$  and  $y_r$  - the reference positions for  $x$  and  $y$ , respectively, produced by the PaPl approach described in the previous section,  $e_x = x_r - x$  and  $e_y = y_r - y$  - the tracking errors for  $x$  and  $y$ , respectively, CB-xp and CB-yp - computation blocks that provide the estimates  $\hat{x}_r$  and  $\hat{y}_r$  of derivatives  $\dot{x}_r$  and  $\dot{y}_r$ , respectively,  $\Delta x_{r,t}$  and  $\Delta y_{r,t}$  - the increments of reference positions  $x_r$  and  $y_r$ , respectively, and  $t$  indicates, as in Section 2, the discrete time index.

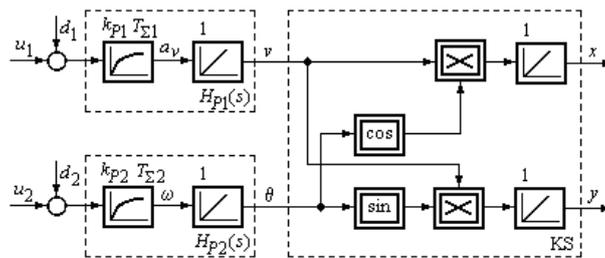


Figure 2: Structure of simplified dynamic model as a controlled process [59].

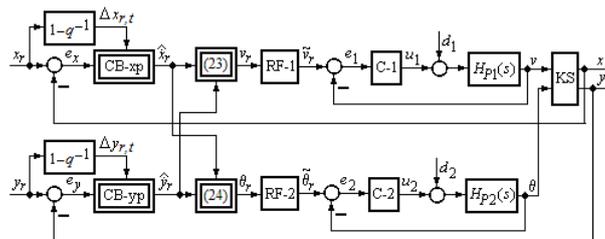


Figure 3: Control system structure for tracking control of NWMRs with two degrees of freedom.

The cascade control system structure given in Figure 3 consists of the two inner loops for  $v$  and  $\theta$ , and the outer loops for  $x$  and  $y$ . The blocks CB-xp, CB-yp in the outer loops operate in terms of

$$\hat{x}_{r,t} = \begin{cases} \Delta x_{r,t}/h & \text{if } |e_{x,t}| \leq \varepsilon_x, \\ e_{x,t}/h & \text{otherwise,} \end{cases} \quad (22)$$

$$\hat{y}_{r,t} = \begin{cases} \Delta y_{r,t}/h & \text{if } |e_{y,t}| \leq \varepsilon_y, \\ e_{y,t}/h & \text{otherwise,} \end{cases}$$

where  $h$  is the sampling period,  $q_1$  is the backward shift operator,  $\Delta x_{r,t} = x_{r,t} - x_{r,t-1}$ ,  $\Delta x_{y,t} = y_{r,t} - y_{r,t-1}$ , the parameters  $\varepsilon_x > 0$  and  $\varepsilon_y > 0$  are the maximum accepted absolute values of the tracking errors corresponding to the positions  $x$  and  $y$ , respectively.

The first two equations in (20) are used to compute the reference inputs  $v_r$  and  $\theta_r$  of the two inner control loops. However, these equations are modified as follows:

$$v_r = \sqrt{(\hat{x}_r)^2 + (\hat{y}_r)^2}, \quad (23)$$

$$\theta_r = \text{atan}(\hat{y}_r/\hat{x}_r), \quad (24)$$

and the last two equation numbers are inserted in the two nonlinear blocks highlighted in Figure 3.

PI controllers tuned by the ESO method [61] are recommended for the processes illustrated in Figure 2. In addition, in this situation it will be used to produce tracking control inputs applied to NWMR. ESO method ensures a convenient trade-off to control system performance indices (percent overshoot, settling time, rise time, phase margin, etc.) of the linear control system by the choice of just a single design and tuning parameter,  $\beta_{ESO}$ ,  $1 < \beta_{ESO} < 20$ . The transfer functions of the two PI controllers that can be used as C-1 and C-2 in Figure 3 are represented as a unified generic PI controller

$$H_{C-\psi}(s) = k_{c\psi}(1 + sT_{i\psi})/s = k_{C\psi}[1 + 1/(sT_{i\psi})], k_{C\psi} = k_{c\psi}T_{i\psi}, \quad (25)$$

where  $k_{C\psi} > 0$  (or  $k_{c\psi} > 0$ ) is the controller gain,  $T_{i\psi} > 0$  is the integral time constant, and the subscript  $\psi \in \{1, 2\}$  indicates the controller number or the control loop number or the controlled process number via (21).

PI tuning conditions specific to ESO method are

$$k_{c\psi} = 1/(\beta_{ESO\psi} \sqrt{\beta_{ESO\psi} k_{P\psi} T_{\Sigma\psi}^2}), T_{i\psi} = \beta_{ESO\psi} T_{\Sigma\psi}, k_{C\psi} = 1/(\sqrt{\beta_{ESO\psi} k_{P\psi} T_{\Sigma\psi}}), \psi \in \{1, 2\}, \quad (26)$$

and the transfer functions of the reference input filters RF-1 and RF-2, that improve the control system performance, are

$$H_{F-\psi}(s) = 1/(1 + \beta_{ESO\psi} T_{\Sigma\psi} s), \psi \in \{1, 2\}. \quad (27)$$

Takagi-Sugeno PI-fuzzy controllers are next used as C-1 and C-2 to replace the linear PI controllers for further performance improvement. The unified structure and input membership functions of the generic Takagi-Sugeno PI-fuzzy controller is given in Figure 4, where TISO-FC $_{\psi}$  are Two Inputs-Single Output fuzzy controller blocks modelled by nonlinear input-output static maps,  $\Delta e_{\psi,t} = e_{\psi,t} - e_{\psi,t-1}$  is the increment of control error, and  $\Delta u_{\psi,t} = u_{\psi,t} - u_{\psi,t-1}$  is the increment of control signal,  $\psi \in \{1, 2\}$ .

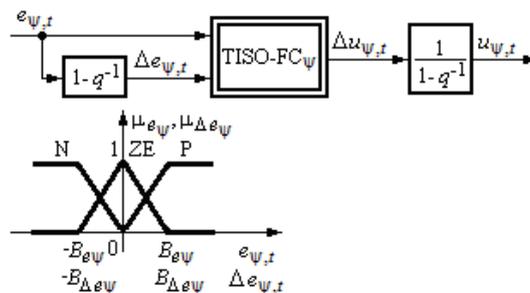


Figure 4: Unified structure and input membership functions of generic Takagi-Sugeno PI-fuzzy controller.

The PI-fuzzy controller design starts with discretizing the continuous-time generic PI controller in (25). Tustin’s method leads to the recurrent equation of generic incremental discrete-time PI controller and its parameters  $KP_{\psi}$  and  $\mu_{\psi}$

$$\Delta u_{\psi,t} = K_{P\psi}(\Delta e_{\psi,t} + \mu_{\psi}e_{\psi,t}), K_{P\psi} = k_{c\psi}(T_{i\psi} - h/2), \mu_{\psi} = 2h/(2T_{i\psi} - h). \quad (28)$$

As shown in [53], [54], [55], [13], the defuzzification in TISO-FC<sub>ψ</sub> blocks is carried out by the weighted average method, and the inference engine makes use of SUM and PROD operators. The generic complete rule base of these blocks consists of only two rules, which contribute to a cost-effective implementation given as follows, where the parameters η<sub>ψ</sub>, 0 < η<sub>ψ</sub> < 1, contribute to the mitigation of control system overshoot:

$$\begin{aligned} &\text{IF}(e_{\psi,t} \text{ IS N AND } \Delta e_{\psi,t} \text{ IS N}) \text{ OR } (e_{\psi,t} \text{ IS P AND } \Delta e_{\psi,t} \text{ IS P}) \\ &\text{THEN } \Delta u_{\psi,t} = \eta_{\psi}K_{P\psi}(\Delta e_{\psi,t} + \mu_{\psi}e_{\psi,t}), \\ &\text{IF}(e_{\psi,t} \text{ IS ZE}) \text{ OR } (e_{\psi,t} \text{ IS N AND } \Delta e_{\psi,t} \text{ IS ZE}) \text{ OR } (e_{\psi,t} \text{ IS N AND } \Delta e_{\psi,t} \text{ IS P}) \\ &\text{OR } (e_{\psi,t} \text{ IS P AND } \Delta e_{\psi,t} \text{ IS ZE}) \text{ OR } (e_{\psi,t} \text{ IS P AND } \Delta e_{\psi,t} \text{ IS P}) \\ &\text{THEN } \Delta u_{\psi,t} = K_{P\psi}(\Delta e_{\psi,t} + \mu_{\psi}e_{\psi,t}). \end{aligned} \quad (29)$$

The application of the modal equivalence principle [18] leads to the tuning equations

$$B_{\Delta e_{\psi}} = \mu_{\psi}B_{e_{\psi}}, \psi \in \{1, 2\}, \quad (30)$$

which are convenient as they lead to the parameter vector  $\rho$  of the two Takagi-Sugeno PI-fuzzy controllers

$$\rho = [\beta_{ESO1} \ B_{e1} \ \eta_1 \ \beta_{ESO2} \ B_{e2} \ \eta_2]^T \in \mathbf{R}^6, \quad (31)$$

and also helps the cost-effective fuzzy controller tuning and implementation.  $T$  in (31) indicates matrix transposition. The OP that ensures optimal tracking is defined as

$$\rho^* = \arg \min_{\rho \in D_{\rho}} J(\rho), J(\rho) = \sum_{t=0}^{\infty} (e_{x,t}^2(\rho) + e_{y,t}^2(\rho)), \quad (32)$$

$\rho^*$  is the optimal parameter vector, i.e., the optimal value of  $\rho$ , and  $D_{\rho}$  is the feasible domain of  $\rho$ .

The GWO-based approach to tracking fuzzy control suggested in this paper consists of steps GWOA1 to GWOA8 presented in the previous section, but with several particular features in these steps described as follows. First, the iteration index  $t$  in the PaPl approach is replaced with  $m$  in the GWO-based approach to tracking fuzzy control, and the maximum number of iterations is  $m_{max}$ . Second, GWO algorithm is mapped onto the OP defined in (32) in step GWOA1 in terms of

$$q = 6, \mathbf{X}_i(m) = \rho, i = 1 \dots N, t_{max} = m_{max}, D_{\mathbf{X}} = D_{\rho}. \quad (33)$$

Third, the performance of each agent is evaluated in step GWOA2 by simulations and/or experiments conducted on the fuzzy control loops. Fourth, the updated vector solution obtained so far  $X_i(m+1) \in D_{\rho}$  is validated in step GWOA6 by checking certain conditions imposed to the fuzzy control loops [53], [54], [55], [13]. Fifth, GWO algorithm is stopped in step GWOA8 and the final solution obtained so far is actually the solution to the OP defined in (32):

$$\rho^* = \arg \min_{i=1 \dots N} J(\mathbf{X}_i(m_{max})). \quad (34)$$

Furthermore, the values of the rest of parameters in the two fuzzy control loops are obtained using (27) and (30).

### 3.1 Real-time experimental validation

The validation of the new GWO-based PaPl and tracking approaches was carried out by a set of experiments conducted on the nRobotic platform using different simple multi-agent settings with at least two NWMRs, different initial robots' positions and target points and accounting for the presence of obstacles. An example of experimental scenario for two NWMRs along the iterations of PaPl approach is illustrated in Figure 5.

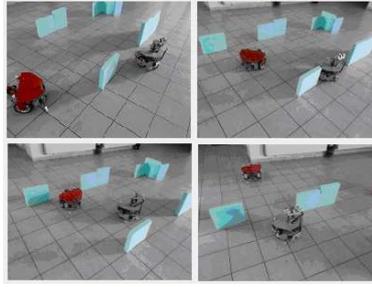


Figure 5: Snapshots along the iterations of the PaPl approach in an example of experimental scenario for two NWMRs and obstacles.

A simplified and representative architecture for this application in nRobotic platform is illustrated in Figure 6. The central component in nRobotic platform is Global path planner, which makes use of different implementations of PaPl including the algorithms presented here. Global path planner has a constraint regarding its application only to environments with an available map; this requires that the SLAM process should have already been applied to the current working environment (real or simulated) and the result is next fed to system via Map component. Robot controllers assigned to individual robots are responsible with the management of resources as sensors and local path planning; local map uncertainties as moving/moved obstacles are considered in this regard. Robot pool manager is responsible with NWMRs management and based on mission profile it can dynamically allocate active robots. Two statically allocated robots (that belong to the DrRobot X80 family) are used in the experimental setup. nRobotic platform supports both simulated and real robots, however combinations of real and simulated robots are not supported.

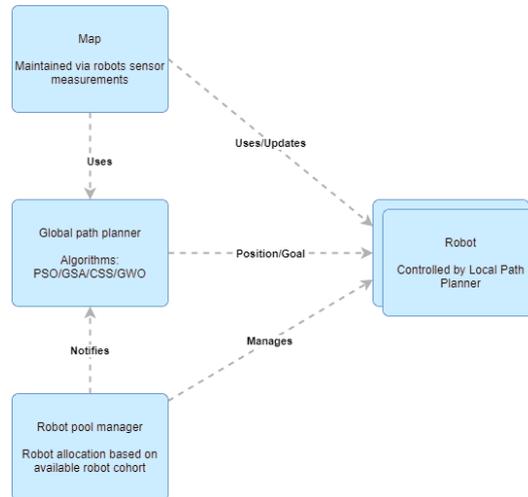


Figure 6: Architectural diagram for NWMRs path planning in nRobotic platform.

The OPs defined in (5) were solved using  $\lambda_1 = 0.4$ ,  $\lambda_2 = 0.3$ ,  $\lambda_3 = 0.3$  for comparison [57], and  $\alpha_{PaPl} = 50$ . For the same reason of comparison, PSO, GSA and CSS algorithms were included in the PaPl approach, with the parameters: for the PSO-based PaPl approach [62]:  $N = 100$ ,  $t_{max} = 30$ , threshold = 10, initial velocity vector  $[-0.2 \ 0.2]^T$ , acceleration constants  $c_1 = 1.49$  and  $c_2 = 1.49$ , linear decrease of inertia weight  $w$  with the advance of PSO algorithm’s iterations within  $w_{min} \leq w \leq w_{max}$ ,  $w_{min} = 0.0001$  and  $w_{max} = 0.5$ , for the GSA-based PaPl approach [63]:  $N = 100$ ,  $t_{max} = 30$ , threshold = 10, and initial velocity vector  $[-0.20.2]^T$ , for the CSS-based PaPl approach [63]:  $N = 100$ ,  $t_{max} = 30$ , initial velocity vector  $[-0.20.2]^T$ , radius of agents = 1, parameter in the denominator of the expression of separation distance between two agents = 0.01, and linear modifications of velocity and acceleration parameters with the advance of CSS algorithm iterations, and the parameters specific to GWO were set to  $N = 100$  and  $t_{max} = 30$ . These parameter values ensure a fair comparison, which is always a problem specific to metaheuristic algorithms as the performance depends on the parameter settings.

The evolutions of the o.f.s defined in (7) are illustrated in Figure 7 for the best agents along the interval  $1...t_{max}$  as an indication of algorithms' convergence. The results correspond to one of the simple experimental scenarios with two robots and one of the populations of agents, i.e.,  $P_{R1}$ . Figure 7 shows that the GWO-based PaPl approach exhibits the fastest convergence. In addition, the execution times of all approaches were measured after running on a computer with Intel I7 @ 2.2 GHz, and the results indicate that PSO is the best one, followed by CSS, GWO and GSA.

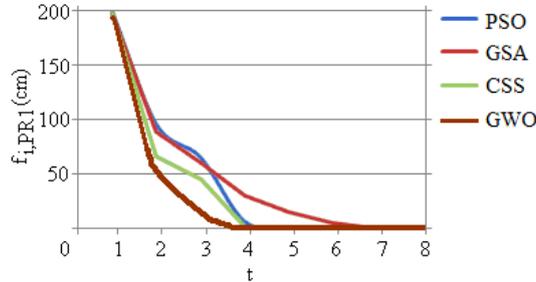


Figure 7: O.f. versus iteration number for four PaPl approaches and best agents in population PR1, with data for PSO, GSA and CSS in [63], [56], [57].

The GWO-based approach to tracking fuzzy control was next validated using the simplified dynamic model (20) of several NWMRs with two degrees of freedom that is integrated in the nRobotic platform, and the control system structure given in Figure 3. Accepting in one of the experimental scenarios that [59] the obstacles are placed at (3, 3), (9, 3), (6, 7), the initial position of the robot is (10, 4), and the goal (or the desired / final position) is placed at (6, 11), one typical reference trajectory  $(x_r, y_r)$  produced by the GWO-based PaPl approach will be presented as follows. All positions are expressed in m.

The values of the parameters in (20) specific to one NWMR are  $k_{P1} = k_{P2} = 1$  and  $T_{\Sigma1} = T_{\Sigma2} = 1$  s. Using the sampling period  $h = 0.01$  s and setting the feasible domain of  $\rho$  and also the search space to

$$D_{\rho} = \{3 \leq \beta_{ESO1} \leq 17\} \times \{20 \leq B_{e1} \leq 40\} \times \{0.55 \leq \eta_1 \leq 0.75\} \times \{3 \leq \beta_{ESO2} \leq 17\} \times \{20 \leq B_{e2} \leq 40\} \times \{0.55 \leq \eta_2 \leq 0.75\}, \quad (35)$$

the GWO-based approach to tracking fuzzy control was applied. The parameters of GWO algorithm were set, such that to reach a good trade-off to convergence and use of allocated resources, to  $N = 20$  and  $m_{max} = 100$ . The o.f.  $J(\rho)$  defined in (32) was evaluated by experiments carried out on a time horizon of 1100 s in terms of the dynamic regime characterized by the reference input trajectory generated by the PaPl approach, zero initial conditions and zero disturbance inputs. A part of the results is exemplified in Figure 8 as the initial robot trajectory (before the application of GWO) and Figure 9 as the final robot trajectory (after the application of GWO). The optimal parameter vector

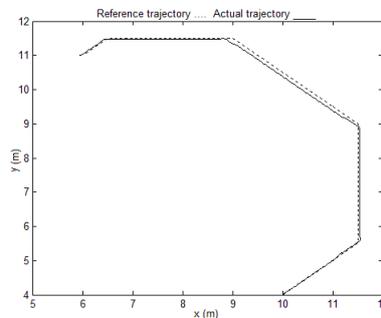


Figure 8: Reference and actual trajectories of NWMR with fuzzy controllers before their GWO-based tuning.

obtained after the  $m_{max} = 100$  iterations of GWO algorithm is

$$\rho^* = [5.0986 \ 39.56 \ 0.7474 \ 16.92 \ 20.08 \ 0.2507]^T, \quad (36)$$

and the performance improvement is illustrated in Figure 9 versus Figure 8 by overall smaller tracking errors. Concluding, the PI-fuzzy controllers produce good tracking control inputs applied to NWMR.

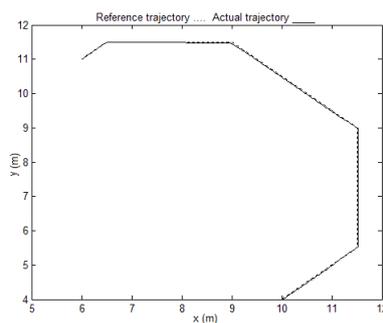


Figure 9: Reference and actual trajectories of NWMR after application of GWO-based approach to tracking fuzzy control.

The minimum o.f. value resulted after running the GWO algorithm is 905465, which is better (lower) compared to the o.f. values resulted after running PSO (909915), GSA (906895) and CSS (906574) algorithms, thus showing the performance improvement. This performance improvement can be called relatively small, but that is normal because the fair comparison with other correctly developed metaheuristic optimization algorithms was carried out.

Nevertheless, the performance improvement can be better understood with the aid of the tracking performance in terms of the above two types of system trajectories, with GWO versus without GWO (illustrated in Figure 9 and Figure 8, respectively), which point out that the performance improvement is major. The conclusions regarding the convergence and execution times are consistent with those formulated for the PaPl approach.

However, these conclusions might be different if other robots in other situations are considered including constraints, which require the adequate modification of the definition of the optimization problem to handle the constraints. Constraints specific to technical and economical operating conditions in terms of various linear and nonlinear systems can be formulated [21], [60], [68] [15], [39], [45], [52]. Knowledge-based and fuzzy systems are included [58], [69], [7], [32].

## 4 Conclusion

This paper proposed two applications of GWO as a representative metaheuristic algorithm. These two applications are organized as two approaches to optimal path planning and optimal tuning of tracking fuzzy controllers for nonholonomic wheeled mobile robots.

Experiments conducted on the nRobotic platform illustrate the performance exhibited by the controllers produced by our approaches, which are built around generic PI and PI-fuzzy controllers, and their potential to be applied as cost-effective control solutions in other applications. Since the experimental setup might be too simple to show the performance difference and improvement offered by our approaches, video demos are available from the authors.

The list of contributions and advantages of the proposed approaches pointed out in Section 1 is not fully discussed and proven in the paper. Future research will be focused on carrying out this discussion associated with deep investigations and comparisons that are important in terms of mentioning shortcomings and limitations. These investigations are currently not included in this version of the paper that is very dense by packing both applications into a single paper. The authors are aware of the fact that a somewhat less dense presentation would significantly increase understandability and readability of the paper and leave room for a more detailed discussion of the approaches, their implementations, and the results. However, the discussion on how GWO performs in comparison with established approaches, which is included in the paper, helps to assess the value of our approaches.

Future research will also be focused on the merge of our path planning and fuzzy controller tuning approaches with classical optimization algorithms [6], [8], [16], [3], [5], [67], and modern ones [42], [4], [48], [1], [26], [51], [66]. The approaches will be generalized such that to work with more compli-

cated tracking fuzzy controller structures, however keeping the objective of cost-effective design and implementation.

## 5 Acknowledgements

This work was supported by grants from the Ministry of Research and Innovation, CNCS - UEFIS-CDI, project numbers PN-III-P1-1.1-PD-2016-0331 and PN-III-P1-1.1-PD-2016-0683, within PNCDI III, and by the NSERC of Canada.

## References

- [1] Abed-alguni, B.H. (2019). Island-based cuckoo search with highly disruptive polynomial mutation, *International Journal of Artificial Intelligence*, 17(1), 57–82, 2019.
- [2] Ajeil, F.H.; Ibraheem, K.I.; Sahib, M.A.; Humaidi, A.J. (2019). Multi-objective path planning of an autonomous mobile robot using hybrid PSO-MFB optimization algorithm, *Applied Soft Computing*, 89, paper 106076, 2020.
- [3] Albu, A.; Precup, R.-E.; Teban, T.-A. (2017). Results and challenges of artificial neural networks used for decision-making in medical applications, *Facta Universitatis, Series: Mechanical Engineering*, 17(4), 285–308, 2019.
- [4] Alvarez Gil, R.P.; Johanyák, Z.C.; Kovács, T. (2018). Surrogate model based optimization of traffic lights cycles and green period ratios using microscopic simulation and fuzzy rule interpolation, *International Journal of Artificial Intelligence*, 16(1), 20–40, 2018.
- [5] Andoga, R.; Fözö, L.; Kovács, R.; Beneda, K.; Moravec, T.; Schreiner, M. (2019). RRobust control of small turbojet engines, *Machines*, 7(3), 1–14, 2019.
- [6] Angelov, P.; Lughofer, E. (2008). Data-driven evolving fuzzy systems using eTS and FLEXFIS: Comparative analysis, *International Journal of General Systems*, 37(1), 45–67, 2008.
- [7] Bădică, A.; Bădică, C.; Ivanović, M.; Dănciulescu, D. (2018). Multi-agent modelling and simulation of graph-based predator-prey dynamic systems: A BDI approach, *Expert Systems*, 35(5), e12263, 2018.
- [8] Baranyi, P.; Korondi, P.; Patton, R.J.; Hashimoto, H. (2004). Trade-off between approximation accuracy and complexity for TS fuzzy models, *Asian Journal of Control*, 6(1), 21–33, 2004.
- [9] Blažič, S. (2014). On periodic control laws for mobile robots, *IEEE Transactions on Industrial Electronics*, 61(7), 3660–3670, 2014.
- [10] Chwa, D.K. (2012). Fuzzy adaptive tracking control of wheeled mobile robots with state-dependent kinematic and dynamic disturbances, *IEEE Transactions on Fuzzy Systems*, 20(3), 587–593, 2012.
- [11] Das, P.K.; Behera, H.S.; Jena, P.K.; Panigrahi, B.K. (2016). Multi-robot path planning in a dynamic environment using improved gravitational search algorithm, *Journal of Electrical Systems and Information Technology*, 3(2), 295–313, 2016.
- [12] Das, P.K.; Behera, H.S.; Jena, P.K.; Panigrahi, B.K. (2016). A hybridization of an improved particle swarm optimization and gravitational search algorithm for multi-robot path planning, *Swarm and Evolutionary Computation*, 28, 14–28, 2016.
- [13] David, R.-C.; Precup R.-E.; Preitl, S.; Szedlak-Stinean, A.-I.; Fedorovici, L.-O. (2018). Application of grey wolf optimization in fuzzy controller tuning for servo systems, In *JTan, Y. (ed.), Swarm Intelligence - Volume 2: Innovation, New Algorithms and Methods*, 363–387, IET Digital Library, London, 363–387, 2018.

- [14] Dewangan, R.K.; Shukla, A.; Godfrey, W.W. (2019). Three dimensional path planning using grey wolf optimizer for UAVs, *Applied Intelligence*, 49(6), 2201–2217, 2019.
- [15] Dragoicea, M.; Dumitrache, I.; Constantin, N. (2015). Adaptive neural control for mobile robots autonomous navigation, *CoRR abs/1512.03351*, 2015.
- [16] Dzitac, I.; Filip, F.G.; Manolescu, M.J. (2017). Fuzzy logic is not fuzzy: World-renowned computer scientist Lotfi A. Zadeh, *International Journal of Computers Communications & Control*, 12(6), 748–789, 2017.
- [17] Fierro, R.; and Lewis, F.L. (1998). Control of a nonholonomic mobile robot using neural networks, *IEEE Transactions on Neural Networks*, 9(4), 589–600, 1998.
- [18] Galichet, S.; Foulloy, L. (1995). Fuzzy controllers: synthesis and equivalences, *IEEE Transactions on Fuzzy Systems*, 3(2), 140–148, 1995.
- [19] Ge, F.-W.; Li, K.; Xu, W.-S.; Wang, Y. (2019). Path planning of UAV for oilfield inspection based on improved grey wolf optimization algorithm, *Proceedings of 2019 Chinese Control and Decision Conference*, Nanchang, China, 1–6, 2019.
- [20] Ge, S.S.; Wang, Z.P.; Lee, T.H. (2003). Adaptive stabilization of uncertain nonholonomic systems by state and output feedback, *Automatica*, 39(8), 1451–1460, 2003.
- [21] Haidegger, T.; Kovács, L.; Preitl, S.; Precup, R.-E.; Benyó, B.; Benyó, Z. (2011). Controller design solutions for long distance telesurgical applications, *International Journal of Artificial Intelligence*, 6(S11), 48–71, 2011.
- [22] Hidalgo-Paniagua, A.; Vega-Rodríguez, M.A.; Nieves Pavón, J.F. (2017). Solving the multi-objective path planning problem in mobile robotics with a firefly-based approach, *Soft Computing*, 21(4), 949–964, 2017.
- [23] Huang, J.S.; Wen, C.Y.; Wang, W.; Jiang, Z.-P. (2014). Adaptive output feedback tracking control of a nonholonomic mobile robot, *Automatica*, 50(3), 821–831, 2014.
- [24] Jiang, Z.-P; Nijmeijer, H. (1997). Tracking control of mobile robots: a case study in backstepping, *Automatica*, 33(7), 1393–1399, 1997.
- [25] Kolmanovsky, H.; McClamroch, N.H. (1995). Developments in nonholonomic control systems, *IEEE Control Systems Magazine*, 15(6), 20–36, 1995.
- [26] La Fe-Perdomo, I.; Beruvides, G.; Quiza, R.; Haber, R.; Rivas, M. (2019). Automatic selection of optimal parameters based on simple soft-computing methods: A case study of micromilling processes, *IEEE Transactions on Industrial Informatics*, 15(2), 800–811, 2019.
- [27] Lal, D.K.; Barisal, A.K.; Tripathy, M. (2016). Grey wolf optimizer algorithm based fuzzy PID controller for AGC of multi-area power system with TCPS, *Procedia Computer Science*, 92, 99–105, 2017.
- [28] Lamini, C.; Benhlima, S.; Elbekri, E. (2018). Genetic algorithm based approach for autonomous mobile robot path planning, *Procedia Computer Science*, 127, 180–189, 2018.
- [29] Lee, J.S.; Kim, D.-W. (2016). An effective initialization method for genetic algorithm-based robot path planning using a directed acyclic graph, *Information Sciences*, 332, 1–18, 2016.
- [30] Li, Z.J.; Deng, J. Lu, R.Q.; Xu, Y.; Bai, J.J.; Su, C.-Y. (2016). Trajectory-tracking control of mobile robot systems incorporating neural-dynamic optimized model predictive approach, *IEEE Transactions on Systems, Man, and Cybernetics Systems*, 46(6), 740–749, 2016.

- [31] Lim, R.S.; La, H.M.; Sheng, W. (2014). A robotic crack inspection and mapping system for bridge deck maintenance, *IEEE Transactions on Automation Science and Engineering*, 11(2), 367–378, 2014.
- [32] Liu, H.; Dzitac, I.; Guo, S. (2018). Reduction of conditional factors in causal analysis, *International Journal of Computers Communications & Control*, 13(3), 383–390, 2018.
- [33] Liu, J.H.; Yang, J.G.; Liu, H.P.; Tian, X.J.; Gao, M. (2017). An improved ant colony algorithm for robot path planning, *Soft Computing*, 21(9), 5829–5839, 2017.
- [34] Low, E.S.; Ong, P.; Cheah, K.C. (2019). Solving the optimal path planning of a mobile robot using improved Q-learning, *Robotics and Autonomous Systems*, 115, 143–161, 2017.
- [35] Ma, Y.; Wang, H.; Xie, Y.; Guo, M. (2014). Path planning for multiple mobile robots under double-warehouse, *Information Sciences*, 278, 357–379, 2014.
- [36] Mac, T.T.; Copot, C.; Tran, D.T.; De Keyser, R. (2016). Heuristic approaches in robot path planning: A survey, *Robotics and Autonomous Systems*, 86, 13–28, 2016.
- [37] Mac, T.T.; Copot, C.; Tran, D.T.; De Keyser, R. (2017). A hierarchical global path planning approach for mobile robots based on multi-objective particle swarm optimization, *Applied Soft Computing*, 59, 68–76, 2017.
- [38] Melin, P.; Astudillo, L.; Castillo, O.; Valdez, F.; Garcia, M. (2013). Optimal design of type-2 and type-1 fuzzy tracking controllers for autonomous mobile robots under perturbed torques using a new chemical optimization paradigm, *Expert Systems with Applications*, 40(8), 3185–3195, 2013.
- [39] Michail, K.; Deliparaschos, K.M.; Tzafestas, S.G.; Zolotas, A.G. (2016). AI-based actuator/sensor fault detection with low computational cost for industrial applications, *IEEE Transactions on Control Systems Technology*, 24(1), 293–301, 2016.
- [40] Mirjalili, S.; Mirjalili, S.M.; Lewis, A. (2014). Grey wolf optimizer, *Advanced Engineering Software*, 69, 46–61, 2014.
- [41] Mitić, M., Vuković, N., Petrović, M., and Miljković, Z. (2018). Chaotic metaheuristic algorithms for learning and reproduction of robot motion trajectories, *Neural Computing and Applications*, 30(4), 1065–1083, 2018.
- [42] Mls, K.; Cimler, R.; Vaščák, J.; Puheim, M. (2017). Interactive evolutionary optimization of fuzzy cognitive maps, *Neurocomputing*, 232, 58–68, 2017.
- [43] Mohanta, J. C.; Keshari, A. (2019). A knowledge based fuzzy-probabilistic roadmap method for mobile robot navigation, *Applied Soft Computing*, 79, 391–409, 2019.
- [44] Nazarahari, M.; Khanmirza, E.; Doostie, S. (2019). Multi-objective multi-robot path planning in continuous environment using an enhanced genetic algorithm, *Expert Systems with Applications*, 115, 106–120, 2019.
- [45] Nikoobin, A.; Habibollahi, A.; Dideba, A. (2018). Lyapunov-based nonlinear disturbance observer for n-links flexible joint robot manipulators. *Control Engineering and Applied Informatics*, 20(2), 22–32, 2018.
- [46] Noshadi, A.; Shi, J.; Lee, W.S.; Shi, P.; Kalam, A. (2016). Optimal PID-type fuzzy logic controller for a multi-input multi-output active magnetic bearing system, *Neural Computing and Applications*, 27(5), 2031–2046, 2016.
- [47] Ojha, V.; Abraham, A.; and Snášel, V., (2019). Heuristic design of fuzzy inference systems: A review of three decades of research, *Engineering Applications of Artificial Intelligence*, 85, 845–864, 2019.

- [48] Osaba, E.; Del Ser, J.; Sadollah, A.; Bilbao, M.N. Camacho, D. (2018). A discrete water cycle algorithm for solving the symmetric and asymmetric traveling salesman problem, *Applied Soft Computing*, 71, 277–290, 2018.
- [49] Park, B.S.; Yoo, S.J.; Park, J.B.; Choi, Y.H. (2010). A simple adaptive control approach for trajectory tracking of electrically driven nonholonomic mobile robots, *IEEE Transactions on Control Systems Technology*, 18(5), 1199–1206, 2010.
- [50] Patle, B.K.; Parhi, D.R.K.; Jagadeesh, A.; Kashyap, S.K. (2019). Application of probability to enhance the performance of fuzzy based mobile robot navigation, *Applied Soft Computing*, 75, 265–283, 2019.
- [51] Petrović, G., Mihajlović, J.; Čojbašić, Ž.; Madić, M.; Marinković, D. (2019). Comparison of three fuzzy MCDM methods for solving the supplier selection problem, *Facta Universitatis, Series: Mechanical Engineering*, 17(3), 455–469, 2019.
- [52] Pozna, C.; Precup, R.-E. (2018). An approach to the design of nonlinear state-space control systems, *Studies in Informatics and Control*, 27(1), 5–14, 2018.
- [53] Precup, R.-E.; David, R.-C.; Petriu, E.M.; Szedlak-Stinean, A.-I.; Bojan-Drăgos, C.-A. (2016). Grey wolf optimizer-based approach to the tuning of PI-fuzzy controllers with a reduced process parametric sensitivity, *IFAC-Papers Online*, 48, 55–60, 2016.
- [54] Precup, R.-E.; David, R.-C.; Petriu, E.M. (2017). Grey wolf optimizer algorithm-based tuning of fuzzy control systems with reduced parametric sensitivity, *IEEE Transactions on Industrial Electronics*, 64 (1), 527–534, 2017.
- [55] Precup, R.-E.; David, R.-C.; Szedlak-Stinean, A.-I.; Petriu, E.M.; Drăgan, F. (2017). An easily understandable grey wolf optimizer and its application to fuzzy controller tuning, *Algorithms*, 10(2), 1–15, 2017.
- [56] Precup, R.-E.; Petriu, E.M.; Fedorovici, L.-O.; Radac, M.-B.; Drăgan, F. (2014). Multi-robot charged system search-based optimal path planning in static environments, *Proceedings of 2014 IEEE Multi-Conference on Systems and Control*, Antibes, France, 1912–1917, 2014.
- [57] Precup, R.-E.; Petriu, E.M.; Radac, M.-B.; Voisan, E.-I.; Drăgan, F. (2015). Adaptive charged system search approach to path planning for multiple mobile robots, *IFAC-PapersOnLine*, 48(10), 294–299, 2015.
- [58] Precup, R.-E.; Preitl, S.; Balas, M.; Balas, V. (2004). Fuzzy controllers for tire slip control in anti-lock braking systems, *Proceedings of 2004 IEEE International Conference on Fuzzy Systems*, Budapest, Hungary, 3, 1317–1322, 2004.
- [59] Precup, R.-E.; Preitl, S.; Szabo, C.; Korondi, P.; Szemes, P. (2005). On some low-cost tracking controllers for mobile robots, *Control and Intelligent Systems*, 33(1), 1–12, 2005.
- [60] Precup, R.-E.; Tomescu, M.-L.; Drăgos, C.-A. (2014). Stabilization of Rössler chaotic dynamical system using fuzzy logic control algorithm, *International Journal of General Systems*, 43(5), 413–433, 2014.
- [61] Preitl, S.; Precup, R.-E. (1999). An extension of tuning relations after symmetrical optimum method for PI and PID controllers, *Automatica*, 35(10), 1731–1736, 1999.
- [62] Purcaru, C.; Precup, R.-E.; Iercan, D.; Fedorovici L.-O.; David, R.-C.; Drăgan, F. (2013). Optimal robot path planning using gravitational search algorithm, *International Journal of Artificial Intelligence*, 10(S13), 1–20, 2013.
- [63] Purcaru, C.; Precup, R.-E.; Iercan, D.; Fedorovici, L.-O.; Petriu, E.M.; Voisan, E.-I. (2013). Multi-robot GSA- and PSO-based optimal path planning in static environments, *Proceedings of 9<sup>th</sup> International Workshop on Robot Motion and Control*, Wasowo, Poland, 197–202, 2013.

- [64] Radmanesh, M.; Kumar, M.; Sarim, M. (2018). Grey wolf optimization based sense and avoid algorithm in a Bayesian framework for multiple UAV path planning in an uncertain environment, *Aerospace Science and Technology*, 77, 168–179, 2018.
- [65] Rao, A.M.; Ramji, K.; Kumar, T.N. (2018). Intelligent navigation of mobile robot using grey wolf colony optimization, *Materials Today: Proceedings*, 5(9), 19116–19125, 2018.
- [66] Roman, R.-C.; Precup, R.-E.; Bojan-Dragos, C.-A.; Szedlak-Stinean, A.-I. (2019). Combined model-free adaptive control with fuzzy component by virtual reference feedback tuning for tower crane systems, *Procedia Computer Science*, 162, 267–274, 2019.
- [67] Roman, R.-C.; Precup, R.-E.; Petriu, E.M.; Dragan, F. (2019). Combination of data-driven active disturbance rejection and Takagi-Sugeno fuzzy control with experimental validation on tower crane systems, *Energies*, 12(8), 1548, 1–19, 2019.
- [68] Rotariu, C.; Pasarica, A.; Andruseac, G.; Costin, H.; Nemescu, D. (2014). Automatic analysis of the fetal heart rate variability and uterine contractions, *Proceedings of 8<sup>th</sup> International Conference and Exposition on Electrical and Power Engineering*, Iasi, Romania, 1–6, 2014.
- [69] Saeed, A.A.M.; Dănciulescu, D. (2018). Modern interfaces for knowledge representation and processing systems based on markup technologies, *International Journal of Computers Communications & Control*, 13(1), 117–128, 2018.
- [70] Sahoo, B.P.; Panda, S. (2018). Improved grey wolf optimization technique for fuzzy aided PID controller design for power system frequency control, *Sustainable Energy, Grids and Networks*, 16, 278–299, 2018.
- [71] Tanner, H.G.; Kyriakopoulos, K.J. (2000). Discontinuous backstepping for stabilization of non-holonomic mobile robots, *Proceedings of 2002 IEEE Conference on Robotics and Automation, Washington, DC, USA*, 3948–3953, 2000.
- [72] Yang, Y.J.; Pan, J.; Wan, W.W. (2019). Survey of optimal motion planning, *IET Cyber-systems Robotics*, 1(1), 13–19, 2019.
- [73] Zafar, M.N.; Mohanta, J.C. (2018). Methodology for path planning and optimization of mobile robots: A review, *Procedia Computer Science*, 133, 141–152, 2018.
- [74] Zhang, W.B.; Gong, X.P.; Han, G.; Zhao, Y.T. (2017). An improved ant colony algorithm for path planning in one scenic area with many spots, *IEEE Access*, 5, 13260–13269, 2017.
- [75] Zohar, I.; Ailon, A.; Rabinovici, R. (2011). Mobile robot characterized by dynamic and kinematic equations and actuator dynamics: Trajectory tracking and related application, *Robotics and Autonomous Systems*, 59(6), 343–353, 2011.



Copyright ©2020 by the authors. Licensee Agora University, Oradea, Romania.

This is an open access article distributed under the terms and conditions of the Creative Commons Attribution-NonCommercial 4.0 International License.

Journal's webpage: <http://univagora.ro/jour/index.php/ijccc/>



This journal is a member of, and subscribes to the principles of,  
the Committee on Publication Ethics (COPE).

<https://publicationethics.org/members/international-journal-computers-communications-and-control>

*Cite this paper as:*

Precup, R.-E.; Voisan, E.-I.; Petriu, E. M.; Tomescu, M. L.; David, R.-C. ; Szedlak-Stinean, A.-I.; Roman, R.-C. (2020). Grey Wolf Optimizer-Based Approaches to Path Planning and Fuzzy Logic-based Tracking Control for Mobile Robots, *International Journal of Computers Communications & Control*, 15(3), 3844, 2020.

<https://doi.org/10.15837/ijccc.2020.3.3844>