

SOFTWARE/HARDWARE DESIGN OF DECISION-MAKING CONTROLLERS FOR OBJECT NAVIGATION IN HORIZONTAL PLANE

Mohammed Amine Boulahlib, Momcilo Milinovic, Mourad Bendjaballah, Olivera Jeremic, Nenad Sakan

Original scientific paper

The paper aims to research the orientation possibilities of an object in the horizontal plane, from its start course into a required orientating recourse, by using simplified navigation methods. The object's directed motion uses controlling powering impulses, variable distributed in time, during constrained motion time. Three logical decision-making methods are designed for calculating the best maneuvering trajectory with minimal error on the target. Computing the powering impulses, their execution instances, as well as their types, are ensured by the methods and presented in the paper. The developed controlling methods are: a modified multiple shooting method, a new control law, called in this paper, current error orientation, as well as a fuzzy logic method. These methods are designed as decision-making software implemented in an electronic hardware as a predefined programmable controller. This provides pre-programmable orientation of the object at the very beginning of course motion, towards a targeted point settled out of initial direction. The controlling methods use optimal diversification of full elapsed determining time to execute, in sequences, appropriate types and number of powering impulses. Simulation tests of the methods, as well as the designed hardware, are also presented in this paper as a contribution to the development researches of horizontal motion control.

Keywords: *fuzzy logic controller; guidance laws; programmable motion; shooting method; software/hardware set up*

Dizajn softvera/hardvera kontrolera za donošenje odluke pri navigaciji objekta u horizontalnoj ravnini

Izvorni znanstveni članak

Cilj rada je istražiti mogućnosti orijentacije objekta u horizontalnoj ravnini, počevši od njegovog početnog kursa u zahtjevani rakurs, koristeći pojednostavljene metode navigacije. Usmjeravanje gibanja objekta koristi kontrolirane pogonske impulse neravnomjerno distribuirane u ograničenom vremenskom intervalu gibanja. Dizajnirane su tri metode logičkog odlučivanja za izračunavanje najbolje putanje, čije su greške na cilju minimalne. Računanje pogonskih impulsa, njihovih izvršnih instanci kao i tipova, prezentirani su u ovom radu. Razvijene kontrolne mjere su: modificirana višestruka shooting metoda, odnosno novi zakon upravljanja kako je nazvan u ovom radu, trenutna orijentacijska greška, kao i metoda fuzzy logike. Metode su projektirane kao softver za donošenje odluke implementiran u elektronski hardver kao predefiniрани programabilni kontroler. To daje preliminarno programiranje usmjeravanja objekta na samom početku kursa gibanja prema ciljnoj točki smještenoj van početnog pravca. Metodama se optimiziraju raspodijele ukupno determiniranog vremena radi realiziranja odgovarajućih tipova i broja pogonskih impulsa u sekvencama. Simulacijski testovi ovih metoda, kao i projektirani hardver, također su prezentirani u radu kao doprinos razvojnog istraživanju upravljanja horizontalnim gibanjem.

Ključne riječi: *fuzzy logički kontroler; programabilna gibanja; shooting metoda; softver/hardver oprema; zakoni vođenja*

1 Introduction

Controlled objects in the horizontal plane have been the subject of research in tremendous number of papers over the last decades. The studied controlled objects are mainly wheeled mobile robots and vehicles. Their control presents a serious challenge as it brings laterally all the problems that can be faced in control theory. Various approaches to their control are offered in literature. There are approaches that do not aim to imitate the human behaviour but simply use the well-known control forms. These are time-varying controllers, typically known by low convergence speed [1, 2], or time-invariant controllers, with discontinuities, usually characterized by exponential convergence speed [3, 4]. The approaches to multi-level controllers consist of two levels: the lower one is presented by the classical proportional, integral and differential (PID) laws of control [5], while the higher-sophisticated level, mainly targets an imitation of human behaviour, with a goal to avoid constraints imposed by the environments [6, 7], or with an appropriate selection of the desired robot trajectory [8]. The present research paper considers the navigation problem of a new subject of control, referred as an optimal orientated motion of an object, executed by powering impulses (henceforth impulses), in a horizontal plane, from its start course into a required recourse, orientated to a target point. The object's motion has the next specifications: a rigid constrain of full motion time, only one-way possibility of

maneuver and a limited number of impulses. These specifications are interpreted by the mathematical model but lead to development of new approaches for the object's control.

The paper presents three new applications of controlling approaches: numerical controlling algorithm presented by a Modified Multiple Shooting method (henceforth MMS), time-invariant controller, called in this paper Current Error Orientation (henceforth CEO), as well as fuzzy logic method (henceforth FLC), basically developed as requirement to imitate human behaviour, presented in papers [9, 10]. The goal of these controlling methods is to evaluate the best navigating-maneuvering trajectory to the target point from the initial course, by making decision on the types and sequence of impulses' execution. Derived simulations, supported by numerous optimizing examples with and without presence of perturbations, are discussed further in this paper. The controlling methods are implemented on a software/hardware experimental platform designed for simulating types and execution sequences of the impulses, calculated by the mentioned methods. The research results given in this paper are a new contribution in controlling process, for object's simplified navigation, based on their sequent orientation during elapsed time instants, programmed previously, as designed requirements, to move from starting to the targeting point.

2 Mathematical model of the object motion

It is supposed that the object's motion is achieved in the horizontal plane and constrained by invariant full time τ and same initial distance-to-target with and without maneuver D (Fig. 1a). This motion consists of two phases, first is a straight line phase with constant velocity without any external controlling inputs, whereas the second one is a controlled orientating phase. First phase' course and its radial displacement, before orientating the object are given by equations of motion, in the polar coordinate system, respectively as,

$$\varphi^{(1)} = \varphi_0, \rho^{(1)} = V_0^{(1)}t \tag{1}$$

where φ_0 is the initial course and $V_0^{(1)}$ is the initial object's velocity in the 1st phase.

Straight-line phase is presented in Fig. 1a as the trajectory from A to B with a required duration of $\tau^{(1)} = \tau \times \tau^*$, where $\tau^* = \tau^{(1)} / \tau$ is the relative time factor.

Initial distance-to-target D is presented in Fig. 1a as the distance from A to C (with maneuver) which is equal to the distance from A to C' (without maneuver). The distance D is given as;

$$D = V_0^{(1)}\tau \tag{2}$$

The second orientating phase is mainly defined by two parameters, one is the relative time factor τ^* and the other is turning course angle $\Delta\varphi$ (Fig. 1a). This phase is divided into (N) sub-phases noted as (i+1) for $i = 1, 2, \dots, N$, after the achievement of the first phase. The phase duration is $\tau^{(2)} = \tau - \tau^{(1)}$ along recourse line (BC) (Fig. 1b).

It is supposed that the initial horizontal velocity is known at the moment of maneuvering and that for each of the sub-phases the horizontal velocity is generally expressed as

$$\vec{V}_0^{(i+1)} = \vec{V}_1^{(i)} + \vec{U}^{(i+1)} = \vec{V}_0^{(i)} + \vec{U}^{(i+1)} \tag{3}$$

where $\vec{V}_0^{(i+1)}$ and $\vec{V}_0^{(i)}$ are the initial velocity vectors in the sub-phases (i) and (i+1) respectively, whereas $\vec{U}^{(i+1)}$ represents the orientating velocity vector. It has to be noted that the initial and final velocity vectors in any sub-phases are the same ($\vec{V}_1^{(i)} = \vec{V}_0^{(i)}$).

Initial course and radial distance for each sub-phase (i+1) correspond to their achieved homologues in the previous sub-phase i (Fig. 1b) as,

$$\rho_0^{(i+1)} = \rho_1^{(i)} \text{ and } \varphi_0^{(i+1)} = \varphi_1^{(i)} \tag{4}$$

In this paper, the initial values of the radial unit vector $\vec{\lambda}_0^{(i+1)}$ as well as the turning unit vector $\vec{\mu}_0^{(i+1)}$ in the particular phase (i + 1) are respectively equal to their homologues $\vec{\lambda}_0^{(i)}$ and $\vec{\mu}_0^{(i)}$ at the end of the previous phase (i) (Fig. 1b), as

$$\vec{\lambda}_0^{(i+1)} = \vec{\lambda}_1^{(i)} \text{ and } \vec{\mu}_0^{(i+1)} = \vec{\mu}_1^{(i)} \tag{5}$$

If the duration of a sub-phase is $\tau^{(2)}$, starting at the maneuver beginning B and achieving at target point C, the turning motion has only one sub-phase. This case is considered as mono-phase and its turning distance and time are invariant values in the above model.

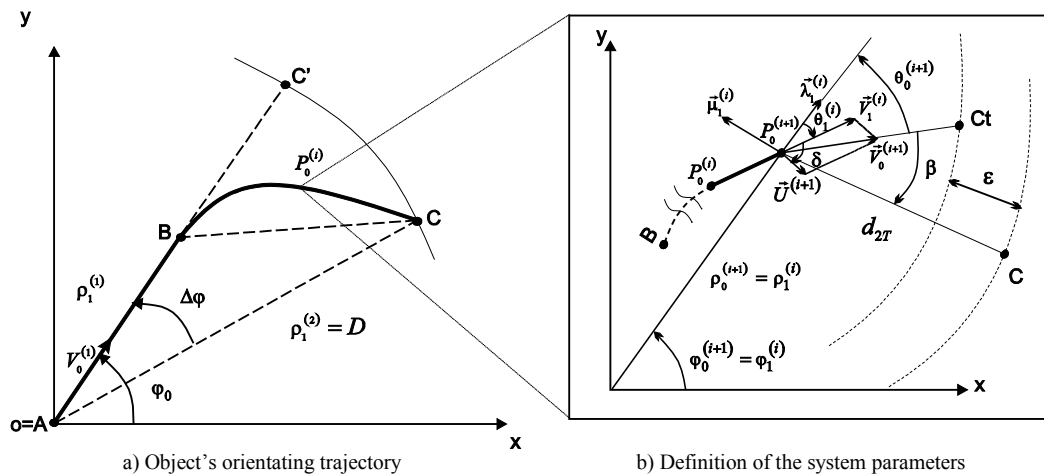


Figure 1 Controlled object trajectory and parameters

Mono-phase turning motion requires higher values of orientating velocity components for the maneuvering motion, while multi-phase turning is less restrictive and can be used to avoid the rigid motion conditions during the orientation. The mono-phase maneuvering path, along recourse line (B to C) (Fig.1-a) is considered as a referencing distance to test achieved maneuvering

recourses in sub-phases, and final motion errors to target C.

The orientating velocity $\vec{U}^{(i+1)}$ in Eq. (3) is characterized by its magnitude $U^{(i+1)}$ and direction $(\nu + \theta_1^{(i)})$, in the polar coordinate system (Fig. 1b), as;

$$\vec{U}^{(i+1)} = U^{(i+1)}[\cos(\nu + \theta_1^{(i)})\vec{\lambda}_0^{(i+1)} + \sin(\nu + \theta_1^{(i)})\vec{\mu}_0^{(i+1)}] \tag{6}$$

where ν is the angle between $\vec{U}^{(i+1)}$ and $\vec{v}_0^{(i)}$, while $\theta_1^{(i)}$ is the final direction of $\vec{v}_0^{(i)}$ in the sub-phase (i). By scalar and vector optimization of the orientating velocity vector given in Eq. (6), three cases of impulses generating the orientating velocity are adopted to be used in simulation tests. They are the next;

- *Lateral impulse for turn-of motion* where the orientating velocity \vec{U} is characterized by a finite magnitude and a fixed angle ν nearby 80° . This type of impulses, noted as \vec{U}_{Lat} , has two components: one is orthogonal to the object's velocity \vec{v}_0 , which represents the only contribution to the turning motion and the major part of orientating velocity magnitude. The other one is longitudinal, as velocity addition along \vec{v}_0 direction, to slightly adjust misses due to lateral motion path. The expression of object's velocity, in the ($i+1$) sub-phase, in the polar coordinate system, is given by :

$$\vec{v}_0^{(i+1)} = \left[V_1^{(i)} \cos(\theta_1^{(i)}) + U_{Lat} \cos(\nu + \theta_1^{(i)}) \right] \vec{\lambda}_0^{(i+1)} + \left[V_1^{(i)} \sin(\theta_1^{(i)}) + U_{Lat} \sin(\nu + \theta_1^{(i)}) \right] \vec{\mu}_0^{(i+1)} \quad (7)$$

- *Longitudinal impulse for run motion* is a contribution to run-to-target motion. In this case the orientating velocity is noted as \vec{U}_{Lon} , and defined by a finite magnitude and a null angle ν . This impulse role is to correct the missed distance-to-target made by elapsed constrained time to move object to the targeted point C. The expression of $\vec{v}_0^{(i+1)}$ is shown as:

$$\vec{v}_0^{(i+1)} = \left[(V_1^{(i)} + U_{Lon}) \cos(\theta_1^{(i)}) \right] \vec{\lambda}_0^{(i+1)} + \left[(V_1^{(i)} + U_{Lon}) \sin(\theta_1^{(i)}) \right] \vec{\mu}_0^{(i+1)} \quad (8)$$

- *No impulses*, this type motion is the most common where the object continues its movement in the same direction with its same velocity magnitude. In this case, the magnitude of the orientating velocity vector is to equal zero, for that the object's velocity is expressed as;

$$\vec{v}_0^{(i+1)} = \left[V_1^{(i)} \cos(\theta_1^{(i)}) \right] \vec{\lambda}_0^{(i+1)} + \left[V_1^{(i)} \sin(\theta_1^{(i)}) \right] \vec{\mu}_0^{(i+1)} \quad (9)$$

The types of the impulses are abbreviated in the following text as Lat, Lon and Ni for lateral, longitudinal and no impulse respectively. By determining the types of the impulses in every sub-phase, the maneuvering phase may be adjusted and corrected in both misses by distance-to-target and by direction.

Orientating velocities, \vec{U}_{Lat} and \vec{U}_{Lon} in this manuscript, are previously defined by determined time profiled values, starting from 0 to U_{max} during a short elapsed instants $\Delta\tau$. Different logic controllers are tested to select the best impulse execution sequences to make software base, which provides minimal errors in the maneuver powered by impulses.

3 Definition of the controlling methods

Three methods are proposed and implemented to optimize the sequent turning trajectories toward the target, by calculating the instants of activation and the types of impulses. The idea is to divide the maneuvering phase time τ_2 into N intervals of Δt , then respectively apply one of the following decision-making methods to activate one of impulse types, by Eq. (7), (8) and (9), at each interval beginning from 1 to N . The goal is to calculate sub-trajectories for each interval, so that the trajectory to the target C is the connection of the N sub-trajectories. The methods that solve the problem are given in the following paragraphs.

3.1 Modified multiple shooting method

The classical single shooting method is a numerical method widely used for solving two-point boundary value problems (BVP), these are mainly differential equations with initial and final values of the solution prescribed, as given in [11]. This method takes its name from varying the initial slope to suggest the trajectory of an object "shot" from the initial point. That initial slope is sought which results in the trajectory "hitting" the target or the final value, reported by [12]. The controlled object's motion phase can be considered as a kind of BVP where the initial and final values are the start-maneuvering point B and the target C respectively. Taking into consideration the specifications of the system, the variation of the initial slope is limited to the mentioned three types of impulses (Lat, Lon and Ni) as determined by Eq. (7), (8) and (9). The controlling method has to evaluate the trajectory between the boundary points B and C.

The modified multiple shooting method (MMS) uses the time decomposition of the problem, together with additional matching conditions at the time interfaces as used in the original direct multiple shooting [13]. Precisely, the time domain is divided into (N) intervals $\Delta t = [t^{i-1}, t^i]$ for $i = 1, 2, \dots, N$, where $\tau^{(2)} = [t^0, t^N]$ and;

$$\tau^{(1)} = t^0 < t^1 < \dots < t^N = \tau \quad (10)$$

The novelty is that the MMS instead of introducing artificial initial values for each time interval, applies three single shots before selecting the best one. The best shot selection is based on an optimality criterion presented by the minimal temporary final error to the target.

$$E_i(I_i, t_{i-1}) = \sqrt{(x_T - x_f^{(i+1)})^2 + (y_T - y_f^{(i+1)})^2} \quad (11)$$

where I_i is the impulse type selected for the sub-phase ($i+1$) and applied at t_{i-1} . The parameters (x_T, y_T) are the target's coordinates while $(x_f^{(i+1)}, y_f^{(i+1)})$ are the temporary final object's coordinates calculated with the remaining time $t_{2go} = \tau_2 - (i-1)\Delta t$ based on the impulse type by Eqs. (7), (8) and (9). The temporary final object's coordinates are given as

$$\begin{cases} x_f^{(i+1)} = \rho_1^{(i)} \cos(\varphi_1^{(i)}) + (V_0^{(i+1)} t_{2go}) \cos(\varphi_1^{(i)} + \theta_0^{(i+1)}) \\ y_f^{(i+1)} = \rho_1^{(i)} \sin(\varphi_1^{(i)}) + (V_0^{(i+1)} t_{2go}) \sin(\varphi_1^{(i)} + \theta_0^{(i+1)}) \end{cases} \quad (12)$$

It is important to emphasize that the $E_N(I_N, t_{N-1})$ represents the final error distance to the target, noted in the following text as Err_{df} .

The MMS presented above aims to evaluate the optimal trajectory between the boundary points B and C that minimizes the cost functional J of the temporary final errors to the target given as;

$$\min J = \sum_{k=1}^N [E_k(I_k, t_{k-1})]^2 \quad (13)$$

The outputs of the MMS are the types of impulses applied at the instances t_{k-1} that generate the optimal trajectory. Making the MMS into practice, for instance, for the j^{th} interval start between 1 to N , the algorithm computes the trajectories with the remaining time, after the three shots. The closest shot to the target is selected as the best shot. The trajectory induced by the best shot is then adopted for the j^{th} interval and its final values are used as initial values for determination of the best shot for the $(j+1)^{th}$ interval. The optimal couple data resulted in this method is used to evaluate the other methods.

3.2 Current error orientation method

The maneuvering object's motion, in this section, is treated as a controlled navigation between start-orientating point B and the target C, taking into consideration the above mentioned specifications of the system. Based on these facts, a logical decision-making controller is designed to ensure minimal error to the target for this type of systems. In this paper, a new time-invariant controller, called Current Error Orientation controller (CEO), is designed and its structure is showed in Fig. 2.

The designed two-input-one-output CEO controller uses as inputs: the error $E_1(t)$ that marks the disparity between the desired angle β_c and the angle $\beta(t)$, and the error $E_2(t)$ which represents the difference between the desired current error ε_c and the actual current error $\varepsilon(t)$ where $\beta(t)$ represents the angle between the object's velocity vector $\vec{v}^{(i)}(t)$ and the current line-to-target $(P_0^{(i+1)}C)$ (Fig. 1-b), while the current error $\varepsilon(t)$ is defined as the difference between the remaining distance to the target d_{2T} and the distance to go d_{2go} , $(P_0^{(i+1)}Ct)$ (Fig. 1b), in the motion direction calculated by $\vec{v}^{(i)}(t)$ and the remaining motion time t_{2go} as,

$$\varepsilon(t) = d_{2T} - d_{2go} = d_{2T} - V^{(i)} t_{2go} \quad (14)$$

The mathematical expressions of $E_1(t)$ and $E_2(t)$ are given by Eq. (15) and Eq. (16) respectively.

$$E_1(t) = \beta_c - \beta(t) \quad (15)$$

$$E_2(t) = \varepsilon_c - \varepsilon(t) \quad (16)$$

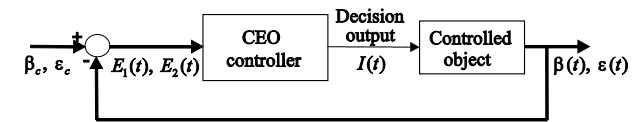


Figure 2 Structure and input/output definition of CEO method

The CEO controller output is the decision $I(t)$ on the type of the impulse to be used (Lat, Lon and Ni). The output $I(t)$ follows a designed time-invariant law given by Eq. (17).

$$I(t) = \begin{cases} Lat & E_1(t) > 0 \\ Lon & E_1(t) \leq 0 \text{ and } E_2(t) > 0 \\ Ni & otherwise \end{cases} \quad (17)$$

The CEO controller given by Eq. (17) regulates the object's direction $\beta(t)$ towards the target, interpreted by $E_1(t)$, by means of lateral impulses (Lat). After achieving the desired direction $\beta(t)$, the CEO controller implicitly adjusts the final error by controlling their current value $\varepsilon(t)$, represented by $E_2(t)$, by executing the longitudinal impulses (Lon). In case of reaching both desired direction $\beta(t)$ and current error $\varepsilon(t)$, the controller makes the decision of no impulse use (Ni), which corresponds to the third type. These logical decisions are generated for each instance t during the maneuvering phase.

The choice of the set-points or the desired inputs is crucial for the reason that it affects directly the work of CEO controller. This choice has to fit the system requirements and specifications. In this application, the choice of β_c and ε_c was $\beta_c \in [2.1^\circ, 2.6^\circ]$ and $\varepsilon_c = 25$ m, which corresponds to the desired requirement for a one-way maneuver.

3.3 Fuzzy logic method

The maneuvering The schematic structure of the decision-making Fuzzy Logic Controller (FLC) is shown in Fig. 3, where FLC is used for controlling the object's lateral motion powered by impulses. FLC is designed, in this present application, to make decisions on the type of impulses to be executed in each time interval. This impulse sequence is made to ensure minimal error for each pre-defined angle $\Delta\varphi$ and relative factor \bar{T} of maneuver.

Membership functions are used by fuzzy logic to define the degree to which crisp physical input values belong to terms in a linguistic variable set. Generally, FLCs are mainly composed of three basic components, which are the fuzzification block, fuzzy inference engine (rule base), and defuzzification block, reported by [14]. The essence of any FLC is the fuzzy inference engine which includes both the decision-making logic and knowledge base. The fuzzification block converts each crisp input information into fuzzy values. The knowledge base comprises a database with necessary linguistic variables as a set of rules, while decision-making logic is used to decide how the fuzzy logic operations are performed. The fuzzy inference engine determines the outputs, as fuzzy values, of each fuzzy IF-THEN rule.

These fuzzy values (outputs) are converted into crispy values in the defuzzification block.

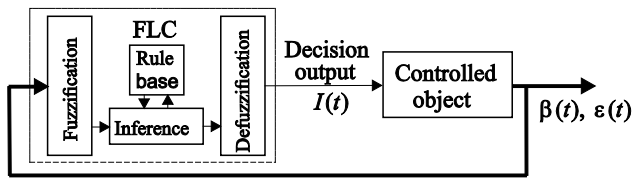


Figure 3 Structure of fuzzy logic method

The FLC designed in this paper is a two-input-one-output controller. The inputs to the FLC are the angle $\beta(t)$ and the current error $\varepsilon(t)$ from the CEO method (Fig. 1b). Both inputs are defined on a universe of discourse with three membership functions, for $\alpha(t)$ (Small, Medium and

Large) and for $Err_{logo}(t)$ (Negative, Zero and Positive) or abbreviated as (S, M and L) and (N, Z and P) respectively.

The output of the FLC is the decision $I(t)$ on the impulse type to be executed at the start of a particular time interval. Accordingly the output is attributed to three membership functions (Lateral, Longitudinal and No impulse), or abbreviated as (Lat, Lon and Ni).

The trapezoidal shaped membership functions are used to fuzzify the input variables while triangular shaped ones are used for output variable. The membership boundaries are shown in Fig.4 and their coefficients $\beta_i, i = 1, \dots, 7, \varepsilon_i, i = 1, \dots, 6, I_i, i = 1, 2, 3$ and w are given in Tab. 1.

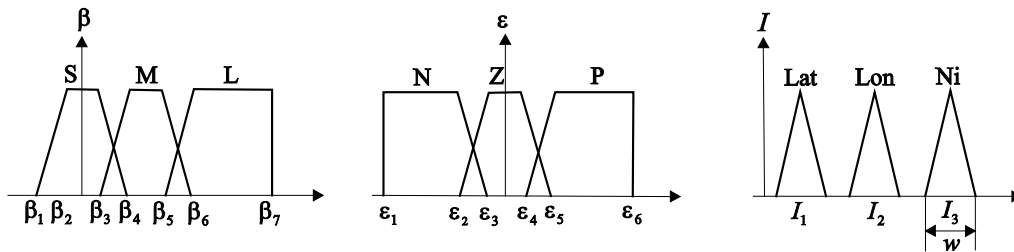


Figure 4 Definition of membership function for input and output variables

Table 1 FLC membership boundaries

Coefficient	β_1	β_2	β_3	β_4	β_5	β_6	β_7	ε_1	ε_2
Value	-7°	$-6,2^\circ$	$2,36^\circ$	$2,64^\circ$	$3,1^\circ$	$3,82^\circ$	70°	$-2E3 \text{ m}$	-150 m
Coefficient	ε_3	ε_4	ε_5	ε_6	I_1	I_2	I_3	w	-
Value	-100 m	100 m	150 m	$4E3 \text{ m}$	0	10	20	8	-

Fuzzy rules given in Tab. 2 are defined basing on the following points:

- Impulses' use causes changes in direction and magnitude of the object's velocity vector depending on their types. Lateral impulses must be used to correct the direction of the velocity vector towards the target;
- In case of a good orientation of controlled object 's velocity vector toward the target, the magnitude of controlled object 's velocity vector rises with the increase in current error $\varepsilon(t)$;
- When the orientation of the controlled object's velocity vector toward the target reaches nearby zero, the need of executing lateral impulses decreases, to respect the requirements for a one-way maneuver.

Table 2 Fuzzy rules set

Rules	$\beta(t)$	$\varepsilon(t)$	$I(t)$
1	-	N	Ni
2	L	-	Lat
3	M	P	Lon
4	M	Z	Lat
5	S	P	Lon
6	S	Z	Ni

The fuzzy inference system presented in this paper is a Mamdani type [10]. Defuzzification method used in this paper is Mean of Maximum (henceforth MoM). The defuzzified value is defined as the mean of all values of the universe of discourse, having maximal membership grades, derived by [15], given by Eq. (18) as;

$$I(t) = \frac{\sum_{j=1}^L c_j}{L} \tag{18}$$

where $I(t)$ is the MoM, c_j is the point at which the memberships function is maximum, and L is the number of times the output distribution reaches the maximum level.

4 Simulation and discussion

To research the ability of the proposed methods to generate maneuvering trajectories with minimal final error, a number of simulations are realized using the above mathematical model. The maneuvers are executed by three types of powering impulses, namely lateral impulses (*Lat*) that generate lateral velocities $U_{Lat} = 21.8 \text{ m/s}$ with optimal angle $\nu = 82^\circ$ and longitudinal impulses (*Lon*) $U_{Lon} = 20.0 \text{ m/s}$ along the object's direction of motion, as well as no impulse (*Ni*) where the controlling velocity is $U_{Ni} = 00.0 \text{ m/s}$. The impulses' execution duration and the sampling time were 300 ms and $\Delta t = 320 \text{ m}$, respectively. For the present study, the target C is situated on a radial distance $D = 36697 \text{ m}$ from the initial location of the object. It is assumed that the object's initial velocity $V_0^{(1)} = 423 \text{ m/s}$ while the full motion time is fixed as $\tau = 86.97 \text{ s}$. To select the final error, the simulation is

repeated for each chosen relative time factor $\tau^* \in [0.5, 0.8]$ and desired turning course angle $\Delta\varphi \in [8^\circ, 18^\circ]$. The simulation results are presented in the Tab. 3. The final error distance to the target Err_{df} is achieved by appropriate number and types of the powering impulses, presented in Tab. 3 as (Lat/Lon) for lateral and longitudinal impulses respectively. The relative time factor τ^* which gives the minimum Err_{df} for each $\Delta\varphi \in [8^\circ, 18^\circ]$ is also shown in Tab. 3.

The comparative analysis shows that the results obtained for all the three methods are overall acceptable. Comparing the results by methods, one can conclude that

all methods give approximately close Err_{df} for each desired couple of maneuvering courses $\Delta\varphi$ and time factor τ^* . The greatest difference in Err_{df} is observed for the couple $\Delta\varphi = 9^\circ$ and $\tau^* = 0.8$, where it is equal to 113.2 m, whereas the lowest is witnessed for the couple $\Delta\varphi = 8^\circ$ and $\tau^* = 0.75$, where it is about 1.66 m. This variation in Err_{df} between methods is due to the differences in the order, and execution instants and the type of impulses calculated by the stated methods.

These simulations are presented for ideal case regardless of all perturbations or mathematical model errors.

Table 3 Simulation results for the decision-making methods

$\Delta\varphi$ (°)	τ^*	MMS method			CEO method			Fuzzy logic method		
		Lat	Lon	Err_{df} (m)	Lat	Lon	Err_{df} (m)	Lat	Lon	Err_{df} (m)
8	0,75	12	1	54,24	12	1	56,9	12	1	55,13
9	0,80	17	4	147,5	18	4	34,3	17	3	124,55
10	0,75	15	3	43,21	15	2	45,06	15	2	39,94
11	0,75	16	3	51,03	16	3	51,03	16	3	59,9
12	0,75	18	4	106,65	19	4	75,26	18	4	127,81
13	0,75	19	5	115,49	20	5	75,75	19	5	116,15
14	0,70	17	3	168,7	17	3	170,49	17	3	168,28
15	0,50	11	0	114,28	11	0	119,19	11	0	138,66
16	0,50	12	0	28,84	12	0	30,42	12	0	48,07
17	0,50	13	1	97,32	13	0	64,29	13	0	64,29
18	0,50	13	1	210,08	14	0	168,05	14	0	168,05

In this sense of meaning, errors Err_{df} presented in Tab. 3 are assumed to be systematic errors (bias). In order to determine the sensitivity of the system due to errors in the model, arbitrary perturbations are added to the system's parameters in each sampling interval. For the reason that the object's velocity is the essence of the mathematical model, the perturbations are expected and chosen to be added to its magnitude as $\Delta V^{(i)} \in [-5, +5] m s^{-1}$, which represents around 25% of orientating velocity magnitude, as well as to its direction as $\Delta\theta_i \in [-0.25^\circ, +0.25^\circ]$. To simulate the effectiveness of the three methods in perturbation presence, the case of $\Delta\varphi = 8^\circ$ and $\tau^* = 0.75$ is selected. The perturbation will be equally added for the three methods in each sampling interval in the maneuvering phase. The simulation is repeated (n) times and the final errors for the three methods obtained in this study follow the normal distributions law. Their resulting parameters are shown in Tab. 4 as the average final errors, $\overline{Err_{df}}$, and the standard deviation σ as well as minimal and maximal error distances $(Err_{df})_{min}$ and $(Err_{df})_{max}$ values.

Table 4 Simulation results in perturbation presence

	$\overline{Err_{df}}$ (m)	σ (m)	$(Err_{df})_{min}$ (m)	$(Err_{df})_{max}$ (m)
MMS	183,98	147,85	5,48	534,6
CEO	178,45	151,78	1,76	577,1
FLC	177,06	150,27	0,43	546,5

4 Hardware design and impulse sequences simulation by the implemented decision-making methods

The static simulation test is realized by software/hardware designed assembly shown in Fig.5. The electronic controller is linked with led diodes which simulate the impulses executions, both in type and instances. The required software control inputs are the desired turning angle $\Delta\varphi \in [8^\circ, 18^\circ]$, relative time factor τ^* as well as one of the controlling methods stated above. The static impulse simulator turns on diodes in time sequences which are the results of the appropriate method given in Tab. 3 for optimal final error distance to the target. In addition to the desired pair $\Delta\varphi$ and τ^* given in Tab. 3, the operator has the possibility to choose the other desired pair for non-optimal final errors. The operator has to choose also the object angular rate (RPM) between a set of predefined ones. This visualization procedure corresponds to the ignition of powering impulses mounted on the controlled object in two rings. First is oriented toward 82° in lateral direction to simulate the lateral impulses work (Lat), whereas the second is mounted in a resulting 0° in axial direction that presents the longitudinal impulses (Lon) Fig. 6. The hardware is designed and programmed to switch on green lights on all the dual-color diodes in order to indicate the system readiness as well as the duration of the first phase $\tau^{(1)}$. To visualize the impulse execution instances and type, the microcontroller turns on red light on the appropriate diode depending on two requirements. First is the motion turning cycle given in Tab. 3, for desired pair of data and chosen controlling method. Second is the diode position on the controlled object ring which

corresponds to predefined chosen rpm (required in real motion). Turning on red light lasts Δt before switching on the mixed light to indicate that impulse is not further available for the turning and lasts all the remaining orientating time t_{2go} .

The electronic input consists of microcontroller unit, the Arduino board, capable of USB connection with the PC and an add-on board. As it could be seen from the block schematics of the hardware presented in Fig.5. The human machine interface based upon character LCD module, accompanied with the tactile switches for selecting the methods and desired pair $\Delta\varphi$ and τ^* . The external EPROM unit enables the storing capability and recalling of large data sets.

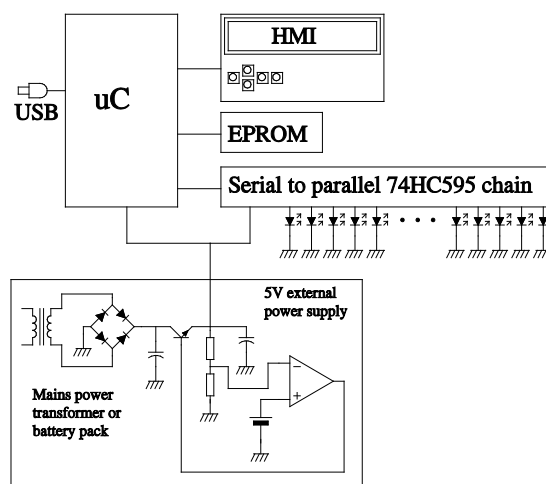


Figure 5 Software /hardware assembly

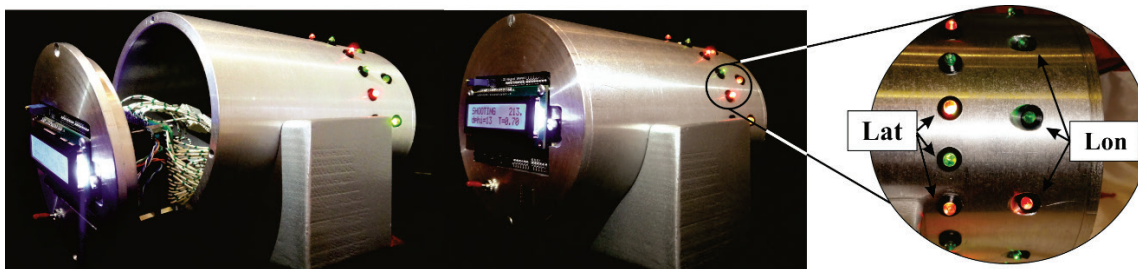


Figure 6 Experimental static simulator of impulses' execution sequences

The visualization process of the impulses' execution is accomplished by means of the 34 LED diodes, 22 for lateral impulses (Lat) and 12 liked simultaneously in 6 pairs for longitudinal impulses (Lon), which corresponds to the maximal number of impulses (Fig. 6). On the applied hardware, the 74HCT595 serial to parallel chain is powered by the external power supply. The diode current is limited by the anode resistor. Since the entire serial chain of 74HCT595 sets the output to the parallel register on strobe of the storage register clock signal (pin 12) all predefined states in serial register appear simultaneously on the output ports. The internal timer interrupt routine enables the precise timing of the visualized process, the 1ms time resolution was selected, and the mentioned hardware possesses the jitter smaller than 62 ns, while the delay is constant and repeatable. The hardware is produced as an add-on board for the Arduino Mega 2560.

5 Conclusion

The comparative analysis of the three decision-making methods, aimed for the maneuvering control of the object, shows promoting and closes results for all methods in presence of external perturbations. Each method has its specifications as the following:

- The modified multiple shooting approach, presented in this paper, combines the direct multiple shooting method and the single one. This constitutes a significant simplification of the maneuvering trajectory optimization and a numerical stability over single shooting methods. The MMS approach involves a minimum logic problem analysis, for that reason it is used as a reference for the preliminary estimation of the controlled objects' parameters. The

disadvantage of this approach is that it does not allow to explicitly including information about the solution into the problem solving procedure.

- The CEO method uses the classical controlling theory by applying a time-invariant controlling law for the multi-input-single-output system. This method presents mathematical solution for maneuvering but requires a further stability analysis before implementing it on a real system.
- The well-known fuzzy method (FLC) has proved its efficiency on numerous systems. It provides an explicitly incorporating knowledge about the solution into the problem by fuzzy rules set.

Further research is orientated on real testing of the designed software/hardware platform mounted on a real mobile system.

Acknowledgements

This research is a part of the PhD applicant's research program and published by the support of the project III-47029MoESTD RS, funded in the year 2016.

6 References

- [1] Pomet, J. B. Explicit design of time-varying stabilizing control laws for a class of controllable systems without drift. // System and Control Letters, 18, (1992), 147-158. [https://doi.org/10.1016/0167-6911\(92\)90019-O](https://doi.org/10.1016/0167-6911(92)90019-O)
- [2] Murray, R. M.; Walsh, G.; Sastry, S. S. Stabilization and tracking for nonholonomic control systems using time-varying state feedback. // IFAC Nonlinear control systems design, (2014), 109-114.

- [3] Astolfi, A. Discontinuous control of the nonholonomic integrator. // *Modeling and Control of Mechanical Systems*, (1997), 293-309. https://doi.org/10.1142/9781848160873_0019
- [4] Kim, B.; Tsiotras, P. Controllers for unicycle-type wheeled robots: Theoretical results and experimental validation. // *Robotics and Automation, IEEE Transactions on*, 18, 3(2002), 294-307. <https://doi.org/10.1109/TRA.2002.1019460>
- [5] Hernández-Guzmán, V. M.; Silva-Ortigoza, R.; Márquez-Sánchez, C. A PD path-tracking controller plus inner velocity loops for a wheeled mobile robot. // *Advanced Robotics*, 29, 16(2015), 1015-1029. <https://doi.org/10.1080/01691864.2015.1033459>
- [6] Wang, M.; Liu, J. N., Fuzzy logic-based real-time robot navigation in unknown environment with dead ends. // *Robotics and Autonomous Systems*, 56, 7(2008), 625-643. <https://doi.org/10.1016/j.robot.2007.10.002>
- [7] Yerubandi, V.; Reddy, Y. M.; Reddy, M. V. K. Navigation system for an autonomous robot using fuzzy logic. // *International Journal of Scientific and Research Publications*, 5, 2(2015).
- [8] Chwa, D. Fuzzy adaptive tracking control of wheeled mobile robots with state-dependent kinematic and dynamic disturbances. // *Fuzzy Systems, IEEE Transactions on*, 20(3), 587-593. (2012). <https://doi.org/10.1109/TFUZZ.2011.2176738>
- [9] Zadeh, L. A. Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic. // *Fuzzy sets and systems*, 90, 2(1997), 111-127. [https://doi.org/10.1016/S0165-0114\(97\)00077-8](https://doi.org/10.1016/S0165-0114(97)00077-8)
- [10] Mamdani, E. H. Advances in the linguistic synthesis of fuzzy controllers. // *International Journal of Man-Machine Studies*, 8, (1976), pp. 669-678. [https://doi.org/10.1016/S0020-7373\(76\)80028-4](https://doi.org/10.1016/S0020-7373(76)80028-4)
- [11] Keller, H. B. *Numerical Methods for Two-point Boundary Value Problems*. Dover, New York. (1992):
- [12] Bailey, P. B.; Shampine, L. F. Waltman, *Nonlinear Two Point Boundary Value Problems*; Press, New York, (1968)
- [13] Bock, H. G.; Plitt K. J. A multiple shooting algorithm for direct solution of optimal control problems. // *Proc. 9th IFAC World Congress Budapest*, Pergamon Press, (1984). [https://doi.org/10.1016/S1474-6670\(17\)61205-9](https://doi.org/10.1016/S1474-6670(17)61205-9)
- [14] Cox, E. Fuzzy fundamentals. // *IEEE Spectrum*, 29, 10(1992), 58-61. <https://doi.org/10.1109/6.158640>
- [15] Iancu, I. A. Mamdani type fuzzy logic controller. INTECH Open Access Publisher. (2012), 331-335, ISBN: 978-953-51-0396-7. <https://doi.org/10.5772/36321>

Nenad Sakan, PhD, research associate

University of Belgrade, Institute of Physics,
Pregrevica 118, 11080, Zemun, Belgrade, Serbia
E-mail: nsakan@ipb.ac.rs

Authors' addresses

Mohammed Amine Boulahlib, PhD applicant

Ministry of Defence of Algeria, PhD applicant at Military Academy, University of Defence, Belgrade
Pavla Jurišića Šturma Street 33, 11000 Belgrade, Serbia
E-mail: m.a.boulahlib@gmail.com

Momcilo Milinovic, PhD, Full professor

University of Belgrade, Faculty of Mechanical Engineering, Kraljice Marije 16, Beograd, Serbia
E-mail: mmilinovic@mas.bg.ac.rs

Mourad Bendjaballah, PhD applicant

Ministry of Defence of Algeria, PhD applicant at Military Academy, University of Defence, Belgrade
Pavla Jurišića Šturma Street, number 33, Belgrade, Serbia
E-mail: mourad.bendjaballa@hotmail.com

Olivera Jeremic, PhD, Associate Professor

University of Belgrade, Faculty of Mechanical Engineering Kraljice Marije 16, Beograd, Serbia
E-mail: ojeremic@mas.bg.ac.rs