UDK 007.5

Krmilniki z mehko logiko
Fuzzy Controllers

ALOJZ HUSSU

0. UVOD

— Aha, končno zelo zanimiv članek!

— Ali ne moreš voziti bolj hitro? -Bi, pa so gume precej izrabljene in cesta tudi ni čisto suha. bova pač malo zamudila.

— Že, toda ta stranka je zelo levo (desno), mislim, da bi bila malo bolj desna (leva) usmeritev bolj primerna.

Tale članek o uporabi mehke logike je pa malo nejasen, se ti ne zdi?

V zgornjih stavkih je veliko pridevnikov in prislovov, s katerimi lahko opisujemo lastnosti določene stvari ali osebe, pa tudi količino nečesa (ni nujno potrebno, da je tisto snovno, npr.: zanimivo, bolj zanimivo, najbolj zanimivo). Količino povemo s stopnjevanjem pridevnika ali prislova ali z drugimi besedami. V splošnem je značilno, da človek nerad uporablja številke za opisovanje količine, raje to opravi bolj opisno, kakor v zgornjih primerih. Pri tem so, vsaj slovnično gledano, če pogledamo pridevnik ali prislov in njuno stopnjevanje, možnosti bolj skromne, saj poznamo samo tri stopnje: osnovnik, primernik in presežnik, če pa dopustimo še stopnjevanje v nasprotni smeri, dobimo skupaj pet stopenj. Seveda si človek, če želi natančneje ločevati, omisli še kakšno drugo lastnost, ki v povezavi s prejšnjo poveča število kombinacij (za dve lastnosti s po 5 stopnjami je 5 × 5 = 25 kombinacij).

Čemu takšen uvod? V zadnjem času je veliko govorjenja o nevronskih sistemih in o sistemih z mehko logiko. Ne eni ne drugi se niso pojavili v zadnjem času; o nevronskih sistemih lahko spremljamo literaturo vsaj od štiridesetih let naprej (definiranje časa s prvo ali drugo vojno je zaradi dogodkov pri nas v zadnjem času postalo nekoliko »mehko« o sistemih z mehko logiko pa od sredine šestdesetih let naprej, ko je Zadeh definiral osnovne pojme in postavil temelje teorije mehkih množic [1], [2], [3]. Če za začetni razvoj sistemov z mehko logiko lahko trdimo, da niso bili vezani na tehnologijo, pa je bil kasnejši razvoj tesneje povezan z uporabo in s tem tudi s tehnologijo, ki v zadnjem času omogoča cenene rešitve in široko uporabo.

Za nevronske sisteme pa lahko trdimo, da so bili v glavnem vedno mišljeni kot neke vrste računalniki, bodisi kot model delovanja človeških možganov oziroma živčnih celic, ali kot sistemi,

0. INTRODUCTION

— Oh, finally a very interesting article!

— Can't you drive a little faster? —I would, but the tires are rather worn out and the road is also not quite dry. We'll come a little later, no problem.

— But, this party is very much on the left (right), I think, a little more right (left) position would be more appropriate.

— This article about fuzzy control is a little fuzzy, don't you think so?

There are a lot of adjectives and adverbs in the above sentences, used to describe properties of a certain thing, person, or the quantity of something (not necessarily material, for example interesting, more interesting, most interesting). The quantity we express with comparison of adjectives and adverbs or with other words. As a general rule, man is uncomfortable with numbers; when he has to tell the quantity, he prefers describing it with words, as in the sentences above. And there are not many possibilities, when we look at the problem grammatically, there are only three comparisons of adjectives and adverbs, positive, comparative and superlative, if comparison in the other, negative direction is taken account of, then we have five steps in total. There are of course more possibilities to obtain or discern more steps. If we take two properties, each with five steps, there are then 25 steps or combnations in total (5 × 5 = 25).

Why such an introduction? There has been a lot of talking about neural systems and fuzzy systems recently. Neither neural nor fuzzy systems are new phenomena. We can follow the literature on neural systems from the forties on (defining time with the first or the second war is getting a little fuzzy, because of recent events in our country), and on fuzzy systems from the midst of the sixties on, when L. Zadeh introduced the first notions and formulated the theory of fuzzy sets [1], [2], [3]. Although the starting period of fuzzy systems was not connected with technology, later development was more strongly connected with applications and for that very reason with technology, which in the last period has enabled cheap solutions and therefore a wide application spectrum.

Neural systems were always intended as some kind of computers, either to model the working of the brain or neurons, or as systems, built to execute certain operations, and for this reason, always strongly connected with technology. The ki naj opravljajo eno ali več operacij, in zaradi tega tesno povezani s tehnologijo. V nadaljevanju se omejimo na sisteme z mehko logiko, in še to samo na krmilne. Najprej nekoliko splošno, potem ob podrobnejšem zgledu. V zgledu bomo obravnavali krmiljenje mobilnega robota na kolesih (WMR). Zgled je povzet po literaturi [8], povzemamo tudi rešitev, da jo lahko primerjamo z rešitvijo, ki jo dobimo po postopku mehke logike.

1. SISTEMI Z MEHKO LOGIKO

V [4] je podan pregled člankov o sistemih z mehko logiko do leta 1985. Zajema 450 referenc, ki so klasificirane po naslednjih področjih uporabe:

- avtomatsko upravljanje
- biologija in medicina
- odločanje
- ekonomija
- inženirstvo
- okolje
- operacijske raziskave
- razpoznavanje vzorcev/klasifikacija
- psihologija
- zanesljivost
- znanost.

Dandanes je uporaba kvečjemu še bolj razširjena: od znanih primerov, predvsem japonskih, o vodenju vlakov, upravljanju pralnih strojev, videokamer, risalnikov itn. [5], do razporejanja ljudi na delovna mesta [6].

Zakaj sistemi z mehko logiko? Veliko je dejavnosti oziroma sistemov, ki zahtevajo avtomatsko upravljanje. Za čimuspejšnejše upravljanje moramo čimbolje poznati objekt upravljanja. To terja njegovo podrobno analizo. Tu se lahko in se tudi pojavljajo problemi, ker ne poznamo dovolj dobro ozadja oz. temeljev, na katerih objekti upravljanja delujejo. Kljub temu lahko včasih nek objekt, ki ga v podrobnostih slabo poznamo, dobro upravljamo, če imamo s sistemom kakšne izkušnje in vemo, kako se obnaša oziroma kako se odziva na določene zunanje vplive. Temu dandanes pogosto pravimo izvedensko znanje. Upravljanje takih objektov, katerih podrobnega delovanja ne poznamo, je eden od razlogov za uporabo sistemov z mehko logiko. Drugi razlog je, da je ta pot snovanja sistemov upravljanja včasih hitrejša, ne zahteva zelo kompleksnih pripomočkov, zlasti če so sistemi nelinearni, ali kako drugače zapleteni, in običajna linearna obdelava ni preprosta. Kot tretji razlog lahko sprejmemo resnico, da je tak način dela (kakršnega bomo spoznali) bližji naravi človeka oz. njegovemu načinu razmišljanja (zato tudi takšni uvodni stavki). Pravijo, da človeški možgani uspešneje obdelujejo vzorce [7], digitalni računalniki pa številke.

paper deals with fuzzy systems only as used in control systems design. They are first treated generally, followed by more detailed discussion of an example. The example is the problem of guiding a wheeled mobile robot (WMR), taken from literature [8]. The solution given in [8] is also summarized, to be able to compare it with that obtained by a fuzzy procedure.

1. FUZZY SYSTEMS

In [4] there is a review of the literature on fuzzy systems up to the year 1985. It covers 450 references, grouped into the following fields of applications:

- automatic control
- biology and medicine
- decision making
- economy
- engineering
- environment
- literature
- operations researches
- pattern recognition/classification
- psychology
- reliability
- science.

Today, there are even more applications, notably from Japan, such as train guidance, control of washing machines, video cameras, plotters etc. [5], and even the optimal placement of people to increase job effectiveness [6].

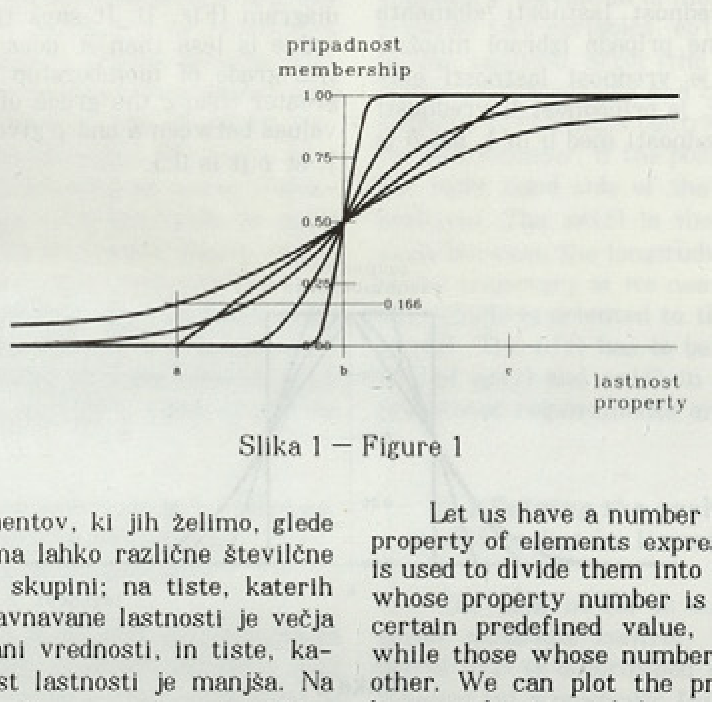
Why fuzzy systems? There are many activities or systems, requiring guidance, regulation, automatic control. For the best control of a system, we should have the best possible knowledge of it. This requires a detailed analysis of the controlled system. Here problems can and do arise when we do not thoroughly understand the inner workings of the system. In spite of this, there are occasions when we can control the system satisfactorily if we have some experience with it and know how it behavies, i. e., how it reacts to certain external inputs. This is nowadays often called expert knowledge. Controlling such systems, whose inner working is not known in detail, is one of the reasons for applying fuzzy systems. Another reason for applying this method of control systems design lies in the fact that it is sometimes faster, and does not require sophisticated support, especially when the controlled system is nonlinear or complex in some other way, and the usual linear treatment unsuitable. A third reason is that this kind of work (to be explained) is in some sense closer to the way people think. It has been found that the human brain is more suitable for pattern processing [7], while digital computers are comfortable with numbers.

Čeprav so obravnavani postopki bližji človekovemu načinu razmišljanja, si prizadevamo, da s primerno aparaturno in programsko opremo postopke avtomatiziramo na digitalnih oz. specializiranih računalnikih.

Kaj je bistvo sistemov z mehko logiko? Začeli bomo z množicami. Zadeh je za nasprotje od ostrega ločevanja pripadnosti nekega elementa neki množici vpeljal mehko, postopno prehajanje elementa v množico. Tako npr. bi se najbrž brez velikih težav odločili, ali ima neko vozilo tri ali manj koles ali več ko tri kolesa. S celimi števili je stvar preprosta, prehod je oster, meja je število 3. Ali je ob osmih zvečer dan ali noč, je pa že težja odločitev. Prehod iz dneva v noč si zlahka predstavljamo kot zvezen prehod, ne oster. Podobno je z mnogimi stvarmi, sladkostjo, hitrostjo, gladkostjo površine itn. Odvisnost oziroma pripadnost lahko ponazorimo z diagramom na sliki 1.

However, though the treated procedures are closer to the human way of reasoning, we still try with suitable hardware and sofware to automate the procedures on digital or specialized computers.

What is the essence of fuzzy systems? We will start with the sets. Zadeh introduced a soft membership function, in contrast to the usual sharp one. So, for example, we can easily decide whether a vehicle has three or less wheels or more than three. There is an easy way with integers, the distinction is sharp, the limit is number 3. We have far more trouble in deciding whether there is day or night at 8 o'clock in the evening. There is a soft, not a sharp transition between day and night. With many other things like sweetness, speed, surface smoothness and so on we have similar problems. The transition or membership function is illustrated in figure 1.



Slika 1 — Figure 1

Imamo nekaj elementov, ki jih želimo, glede na neko lastnost, ki ima lahko različne številčne vrednosti, ločiti v dve skupini; na tiste, katerih številčna vrednost obravnavane lastnosti je večja ali enaka neki predpisani vrednosti, in tiste, katerih številčna vrednost lastnosti je manjša. Na vodoravni osi si predstavljamo nanesene številčne vrednosti. Običajno ležijo večje vrednosti bolj proti desni. Mejo obeh skupin predstavlja vrednost *b*. Če je vrednost lastnosti elementa večja od *b*, potem element pripada eni od skupin, in sicer izbrani množici, sicer pa ne. Tako ločevanje pomeni oster prehod, kajti obravnavani element je ali pa ni v izbrani skupini. Iz prakse vemo, da meritve običajno dajo le nekaj zanesljivih decimalk. Če smo zelo blizu meje vrednosti *b*, je vprašanje, kje v resnici smo, desno ali levo od nje. Zato je včasih primerno, da ne napravimo ostre ločnice, prehod izpeljemo zvezno, elementi blizu meje deloma pripadajo izbrani množici, deloma pa ne. Govorimo torej o funkciji pripadnosti. Nekaj takšnih funkcij z različno hitrimi prehodi prikazuje diagram na sl. 1. Če izberemo najbolj položno odvisnost, potem

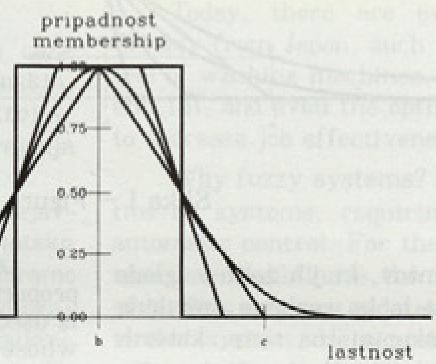
Let us have a number of elements. A certain property of elements expressed as a real number is used to divide them into two groups. Elements, whose property number is greater or equal to a certain predefined value, belong to one group, while those whose number is less belong to the other. We can plot the property numbers on a horizontal axis, with larger values placed more to the right. The dividing number is marked with *b*. If the property value of the chosen element is greater than *b*, then the element belongs to the set, otherwise not. This method of division represents a sharp distinction. We know from practice that measurements give results with only a few reliable decimals. If the results of the measurements are close to the number that marks the dividing line, we cannot be sure to which side they belong, to the left or to the right.

Thus, it is sometimes better not to make a sharp division, but a soft, continuous one. The elements near the dividing line belong partly to the set, partly they do not. In this case we talk about a membership function. A few of them are displayed on the graph (Fig. 1). If we select the curve with the least slope, then we can say for an element which has a property value around *a*, lahko rečemo za element, katerega lastnost, ki jo opazujemo, ima vrednost okoli *a*, da je pripadnost tega elementa množici približno 0,17 in pripadnost elementom, ki niso v množici, približno 0,83 (1—0,17). Pripadnost izrazimo s številko, ki ima vedno vrednost med 0 in 1.

Kot funkcijo pripadnosti si lahko glede na merilo marsikaj izmislimo, da ima le vrednost med 0 in 1. Kljub temu, da operiramo z različnimi pogojnimi stavki in, kakor bomo kasneje videli, z besednimi spremenljivkami, so še vedno potrebne določene računske operacije. Njihovo izvajanje bi bilo pri zelo neprimerno izbranih funkcijah pripadnosti težavno, zato je pametno izbrati preproste odvisnosti. Torej, če oster prehod ni dober, napravimo prehod zvezen, preprost zvezen prehod pa pomeni linearno funkcijo (matematično jo je preprosto opisati). Primer take je tudi na diagramu (sl. 1). Če je npr. vrednost lastnosti elementa manjša od *a*, element ne pripada izbrani množici (pripadnost je 0). Če je vrednost lastnosti elementa večja ali enaka *c*, je pripadnost 1, vrednosti med *a* in *c* pa dajo pripadnosti med 0 in 1, pri *b* je pripadnost 0,5.

that it belongs to the set with a grade of membership equal to 0.17 and that it belongs to the elements not in the set with a grade of membership equal to 0.83 (1 — 0.17). Grade of membership is always given by a number between 0 and 1.

According to different criteria, we can invent quite different membership functions, the only limitation being that the function has to have values between 0 and 1. However, in spite of the fact that we work with different conditional statements and so called linguistic variables we still have to perform some computations. For some strange membership functions the computations can be very complicated, and so, to simplify matters, it is advisable to select simple membership functions. Thus, when a sharp transition is not satisfactory, we introduce a soft, continuous transition. A simple continuous transition represents a linear function (simple to define mathematically). One such function is plotted on the diagram (Fig. 1). It says that an element whose value is less than *a*, does not belong to the set (its grade of membership is 0), if the value is greater than *c* the grade of membership is 1, the values between *a* and *c* give grades between 0 and 1, at *b* it is 0.5.



Slika 2
Figure 2

Podobno velja, če množico definiramo z dvema vrednostma, s spodnjo in zgornjo mejo (sl. 2). Mejo lahko postavimo ostro ali postopno. Elementi, ki imajo vrednost obravnavane lastnosti okoli *b*, pripadajo množici. Bolj ko se oddaljujemo od vrednosti *b* (levo ali desno, proti manjšim ali večjim vrednostim), se pripadnost zmanjšuje. Za funkcijo pripadnosti lahko vzamemo gladko krivuljo, lahko izberemo trapez, trikotnik, ali kaj drugega. Kakor smo že omenili, ima linearna odvisnost prednosti, zato izberemo za nadaljnjo obravnavo trikotnik (podobno kakor trikotnik je tudi trapez sestavljen iz po odsekih linearnih funkcij, vendar je trikotnik preprostejši). Tu bomo končali splošno obravnavo, ob zgledu bomo obdelali podrobnosti.

The same can be said when the set is defined by an upper and lower limit (Fig. 2). The transition can be sharp or gradual. Elements with values around *b* belong to the set, the greater the distance from *b* (on either side), the less is the grade. As a membership function, we can choose a smooth surve, a trapeze, a triangle or something else. As mentioned above, a linear relationship has some advantages, so we choose a triangle for further treatment. A trapeze also has segments which are linear functions, but a triangle is obviously simpler. Here we will stop the general discussion, details will be given in a case problem.

2. PROBLEM KRMILJENJA MOBILNEGA ROBOTA

V [8] je opisan problem krmiljenja mobilnega robota na treh kolesih (eno spredaj in dve zadaj). Sprednjemu kolesu lahko spreminjamo smer in s tem usmerjamo vozilo. Gnani sta zadnji kolesi, hitrost gibanja vozila v smeri vzdolžne osi vozila je konstantna. Problem, zastavljen v [8], je naslednji:

Mobilni robot se mora gibati po vnaprej predpisani trajektoriji. Usmerjamo ga s krmiljenjem sprednjega kolesa, in sicer spreminjamo kot $u(t)$; to je kot, ki ga oklepa vzdolžna os vozila in ravnina, v kateri se vrti prednje kolo (ali kot med pravokotnico na vzdolžno os vozila in osjo sprednjega kolesa). Os vrtenja sprednjega kolesa je vedno vzporedna z vodoravno ravnino, po kateri se robot giblje. Želimo, da je odstopanje od zahtevane trajektorije čimmanjše. Odstopanje od trajektorije sproti merimo in opišemo z dvema spremenljivkama, $ep(t)$ in $es(t)$, tu $ep(t)$ pomeni napako lege vozila (kolikršna je v trenutku t razdalja med središčem vozila — za središče vozila je izbrano njegovo težišče — in najbližjo točko na želeni trajektoriji; če je vozilo desno od trajektorije, je $ep(t)$ pozitiven, $es(t)$ pa pomeni napako smeri vozila (kolikršen je v trenutku t kot med vzdolžno osjo vozila in tangento na trajektorijo v najbližji točki trajektorije; če je vozilo usmerjeno desno je $es(t)$ pozitiven). Določiti je treba $u(t)$ kot funkcijo $ep(t)$ in $es(t)$ tako, da bodo izpolnjene vnaprej izbrane zahteve.

2.1 Rešitev problema krmiljenja mobilnega robota po postopku linearnega kvadratičnega upravljanja

Naloga je v [8] rešena po postopku linearnega kvadratičnega upravljanja. Krmilnik oziroma $u(t)$ določimo tako, da ima integral (pod integralom je kvadratna funkcija):

$$I = \int_0^\infty \left(q_1 \cdot ep(t)^2 + q_2 \cdot es(t)^2 + R \cdot u(t)^2\right) dt$$

za primerno izbrane utežne faktorje q_1, q_2 in R (pozitivna števila) minimalno vrednost. Seveda moramo za sistem, ki ga upravljamo, poznati povezave med $u(t)$ in $ep(t)$ in $es(t)$ oziroma tiste spremenljivke sistema, ki določajo obe napaki. V [8] so te povezave določene z matematičnim modelom vozila. Model je sicer nelinearen, vendar ga pod določenimi pogoji lahko lineariziramo. Za izbrane parametre vozila so bile določene naslednje štiri diferencialne enačbe prvega reda, ki jih zapišemo v matrični obliki:

2. THE PROBLEM OF GUIDING A MOBILE ROBOT

The problem of guiding a mobile robot with three wheels, one steering wheel in front and two driving wheels behind is given in [8]. The speed of the vehicle in a longitudinal direction is constant. The problem as posed in [8] is the following: the mobile robot has to move on a predefined trajectory. It is guided by the front wheel, by changing the angle $u(t)$, the angle between the longitudinal axis of the vehicle and the plane in which the wheel rotates (the angle between the lateral axis of the vehicle and the axis of the front wheel). The axis of the front wheel is always parallel with the horizontal plane on which the robot moves. It is desirable to ensure the least possible deviation from the required trajectory. Deviation is measured continuously and given by two variables, $ep(t)$ and $es(t)$. The $ep(t)$ is the position error (the distance between the centre of the vehicle and the nearest point on the trajectory, the mass centre of the vehicle is selected as its center; if the position of the centre is on the right hand side of the trajectory, then ep is positive). The $es(t)$ is the orientation error (the angle between the longitudinal axis and the tangent to the trajectory at its nearest point, at time t; if the vehicle is oriented to the right, then es is positive). The $u(t)$ has to be determined as a function of $ep(t)$ and $es(t)$ in such a way that some predefined requirements are fulfilled.

2.1 Solving the problem of guiding a mobile robot by linear quadratic control

The task in [8] is solved in the realms of linear quadratic control. The controller or the signal $u(t)$ is determined in such a way that the integral (of a quadratic function):

has a minimal value for some suitably selected factors q_1, q_2 and R (positive numbers). Evidently, we have to know the relationships among $u(t)$ and $ep(t)$ and $es(t)$, or those variables which determine the two errors. In [8] the relationships are given as a mathematical model of the system. The model is nonlinear but can be linearized under certain conditions. The following four linear differential equations written in matrix form were determined for the selected parameters of the vehicle:

$$\dot{\boldsymbol{x}}(t) = A \cdot \boldsymbol{x}(t) + \boldsymbol{b} \cdot u(t),$$

kjer pomeni $\boldsymbol{x}(t)$ vektor stanja, ki ga sestavljajo štiri komponente, in sicer $ep(t)$, $vp(t)$, $\omega(t)$ in $es(t)$, kjer $ep(t)$ in $es(t)$ že poznamo. Vozilo se giblje z neko hitrostjo, ki jo razstavimo na dve komponenti, v smeri vzdolžne osi vozila $vv(t)$ in v smeri pravokotno na vzdolžno os vozila $vp(t)$. Kakor je bilo že rečeno, je vzdolžna hitrost $vv(t)$ konstantna in je vv = 0,4 m/s, tako da je v modelu samo hitrost $vp(t)$. Kotna hitrost $\omega(t)$ je hitrost, s katero se vozilo vrti okoli osi, ki gre skozi središče vozila in je pravokotna na vodoravno podlago, po kateri se giblje.

Matrika A in vektor $\boldsymbol{b}$ v zgornji enačbi imata naslednje vrednosti:

where $\boldsymbol{x}(t)$ is the four dimensional state vector, with components $ep(t)$, $vp(t)$, $\omega(t)$ and $es(t)$. $ep(t)$ and $es(t)$ we already know. The vehicle moves with some speed, which we decompose into two components, one $vv(t)$ in the direction of the longitudinal axis, the other $vp(t)$ in the direction perpendicular to the longitudinal axis. As already stated, the longitudinal speed is constant, $vv(t)$ = = 0.4 m/s, so only the speed $vp(t)$ occurs in the model. The angular speed $\omega(t)$ is the speed of rotation of the vehicle around the axis, going through the centre of the vehicle and perpendicular to the horizontal plane, on which it moves.

Matrix A and vector $\boldsymbol{b}$ in the equation above have the following values:

$$A = \begin{bmatrix} 0{,}4 & 0{,}0014 & -0{,}0016 & 0 \\ 0 & -0{,}1348 & 0{,}6571 & 0 \\ 0 & 0{,}9908 & 0{,}7538 & 0 \\ 0 & -0{,}01 & -0{,}0018 & 1 \end{bmatrix} \qquad \boldsymbol{b} = \begin{bmatrix} 0 \\ 50 \\ 153{,}4 \\ 0 \end{bmatrix}$$

Analiza lastnih vrednosti sistema pokaže, da ima sistem dve lastni vrednosti v koordinatnem izhodišču $\lambda_1 = \lambda_2 = 0$ in dve precej odmaknjeni od koordinatnega izhodišča $\lambda_3 = -96{,}5$ in $\lambda_4 = -418{,}5$, ustrezni komponenti sta močno dušeni, tako da njun vpliv na dinamiko sistema lahko zanemarimo. Če to storimo, lahko model četrtega reda z odpravo spremenljivk $vp(t)$ in $\omega(t)$ iz vektorja stanja poenostavimo v model drugega reda, ki ima kot vektor stanja komponenti $ep(t)/vv$ in $es(t)$. Nova matrika A in vektor $\boldsymbol{b}$ sta:

Eigenvalue analysis of the system shows there are two eigenvalues in the coordinate origin $\lambda_1 = \lambda_2 = 0$ and two quite far from the origin, $\lambda_3 = -96.5$ and $\lambda_4 = -418.5$. The corresponding components are strongly damped, so their influence on the dynamics of the system can be neglected. If this is done, then leaving out the variables $vp(t)$ and $\omega(t)$ from the state vector, the model is simplified to the second order, with components $ep(t)/vv$ and $es(t)$ in the state vector. The new matrix A and the new vector $\boldsymbol{b}$ are now:

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \qquad \boldsymbol{b} = \begin{bmatrix} 0{,}08 \\ 1 \end{bmatrix}$$

Za ta sistem in zgoraj definiran integralski kriterij, če so vse tri uteži enake 1, dobimo po ustreznem izračunu optimalni krmilnik z naslednjim krmilnim zakonom:

For this system and the integral criterion defined above, when all weights are equal to 1, following the usual procedures we get the optimal controller performing the following control law:

$$u(t) = ep(t) + 1{,}3\ es(t)$$

Podrobnejša izvajanja in poenostavljanje modela najdemo v [8].

Details and model simplification can be found in [8].

2.2 Metoda določitve *u(t)* z mehko logiko

Upravljanje z mehko logiko lahko razdelimo v tri faze:

- ugotavljanje stanja objekta,
- določitev želenih ukrepov,
- izvajanje ukrepov.

Pri tej metodi načrtovanja sistemov upravljanja moramo torej določiti postopke, ki naj se izvajajo v posamezni fazi. Pojdimo po vrsti.

2.2 Fuzzy method of determining *u(t)*

Fuzzy control can be divided into three phases:

- determination of the state of the system,
- determination of the required or desired actions,
- execution of actions.

So a fuzzy method of control has to supply the procedures that have to be executed in particular phases. Let us follow the phases in turn.

2.2.1 *Prva faza*

Ta faza zahteva, da se opravijo meritve stanja objekta. Če poznamo stanje in vemo, kaj hočemo, potem lahko ukrepamo. Pri tem pa se pojavi problem, ker imamo v splošnem zelo veliko število različnih stanj objekta. Ker je stanj veliko, se je težko odločiti, kaj v določenem stanju storiti, kako ukrepati. Če imamo zvezne signale, je očitno število mogočih stanj neskončno veliko, pri 10-bitnem analogno-digitalnem pretvorniku imamo še vedno 1024 različnih vrednosti. Take primere človek težko obvladuje. Res je, da v primeru, ko je objekt linearen in pametno izberemo merila za ocenjevanje sistema upravljanja, lahko dobimo linearno upravljanje, tako kakor v zgornjem primeru, ko je upravljanje $u(t)$ določeno preprosto z množenjem odstopanja sistema od želene vrednosti $ep(t)$ in $es(t)$, ki sta kar stanje objekta, s konstanto (krmilni signal je sorazmeren odstopanju objekta od želenega stanja).

Spomnimo se uvodnih stavkov. Manipuliranje s števili ljudem ne leži (mišljeno je rutinsko, vsakodnevno: če ni vsakodnevno, je seveda lahko zelo zanimivo); saj če spremljamo razvoj računalnikov, vedno lahko najdemo v ozadju željo skonstruirati stroj, ki bi namesto človeka izvajal računske operacije. Razpoznavanje vzorcev, če njihovo število ni preveliko, pa je bolj zanimiva dejavnost. In to je bistvo prve faze oziroma splošno uporabe mehke logike. Kako ugotoviti stanje? Res je, da ga izmerimo, toda kombinacij je preveč. Kako zmanjšati število mogočih stanj? Območja vrednosti posameznih spremenljivk, ki jih merimo, razdelimo na končno število (in to majhno) podintervalov (ki so enaki ali različni), in potem samo pogledamo, na katerem od podintervalov je trenutna vrednost posamezne spremenljivke. Če imamo dve spremenljivki in območje vsake razdelimo na pet podintervalov, imamo vsega 25 različnih stanj. Če pa so spremenljivke zvezno spremenljive, je število mogočih stanj neskončno. V primeru, da imamo malo mogočih stanj, je problem snovanja upravljanja laže obvladljiv.

Predpostavimo uporabo računalnikov in s tem analogno-digitalnih pretvornikov, npr. 10-bitnih, pri pretvorbi zveznih vhodnih spremenljivk v dvojiška 10-mestna števila. Če odrežemo od dvojiških števil spodnjih 8 bitov, ostaneta samo dva. Če gledamo samo ta dva bita, imamo vsega štiri različne kombinacije. Celotno področje vrednosti spremenljivke smo razdelili na 4 podpodročja. Kombinacija dveh bitov, ki sta ostala, pove, na katerem podpodročju je trenutna vrednost spremenljivke. Kombinacijo bitov lahko uporabimo za označbo podpodročja (kombinacije so 00, 01, 10, 11). Vendar gremo običajno še dlje, za označbo posameznega

2.2.1 *First phase*

This phase requires measurement of the state of the controlled system. If we know its state and know what we want, then we can act. The problem that generally arises in such cases is that we are dealing with a large number of different states. Consequently it is difficult to decide what to do in a particular state. With continuous variables, there is obviously an infinite number of possible states, with a 10-bit analog to digital converter we still have 1024 different values. Problems of this kind are not to man's measure. If the system is linear and suitable evaluation criteria have been chosen we can get linear control, as was the case in the problem above, where $u(t)$ was simply determined by multiplying the deviation of the system $ep(t)$ and $es(t)$ (which equals the state of the system in our case) from the desired value by a constant vector (the control signal is proportional to the deviation of the system from the desired state).

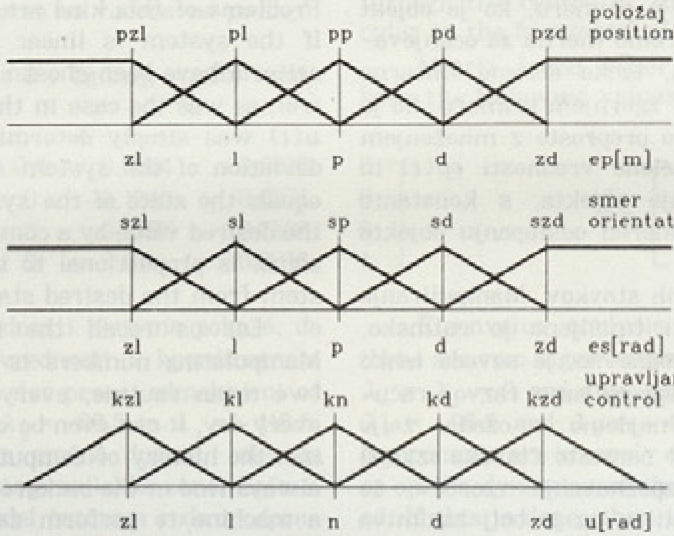
Let us recall the introductory sentences. Manipulating numbers is not man's strong side (we mean routine, everyday work; if it is not every day, it can even be entertaining). If we look into the history of computer development, we can always find in the background a wish to construct a machine to perform calculations instead of a man. But pattern recognition, if their number is not too great, is a more interesting activity. This is the essence of the first phase and generally of the fuzzy method. How to determine the state? We can measure it, but there are too many combinations. How to reduce their number? The allowable regions for particular variables, which are measured, are divided into a finite number (a small one) of subregions or subintervals (equal or different), and then we determine on which subinterval the instantaneous value of a particular variable lies. If we have two variables, and we have five subintervals for each variable, then in total we have 25 different combinations. If we have a small number of states the problem is easier to manage.

Let us assume the use of digital computers and therefore the use of analog-digital converters, say 10-bit, converting a continuous input signal into a stream of 10-bit binary numbers. Cutting the lower 8 bits from binary numbers away, there remain only 2 bits. With 2 bits we have at most 4 different combinations. That way we have divided the whole region for a particular input variable into 4 subregions. The combination of the remaining 2 bits tells us in which of the subregions the instantaneous value of the variable lies. We can use the combinations of two bits as labels for podpodročja ne uporabljamo številk, ampak imena, besedne spremenljivke. Pravzaprav je tudi številka ime, bolj smo pa vajeni imen, ki jih sestavljamo s črkami. Po tej razlagi bi seveda lahko sklepali, da morajo biti podpodročja enaka, lahko so, ni pa nujno potrebno. Lahko so poljubno izbrana, dobro pa je, če imamo razloge za delitev, kakršno napravimo.

Lotimo se torej problema določitve $u(t)$, ki ga moramo rešiti. Prva faza pomeni meritev stanja objekta, ki je v našem primeru podan z $ep(t)$ in $es(t)$ ter določitev položaja, usmeritve in ustreznih funkcij pripadnosti.

subregions (combinations are 00, 01, 10, 11). However, we often employ names (linguistic variables) instead of numbers.. Certainly a number is also a name, but we are more used to write names with letters. Following this discussion, one might assume that the subintervals have to be equal. They can be, but need not necessarily be so. We can chose them deliberately, but it is good to have some reason for the division we make, whatever it is.

Let us start with the problem we have to solve. The first phase represents measurement of the systems state, the variables $ep(t)$ and $es(t)$



Slika 3 — Figure 3

Na sliki 3 so trije diagrami. Na zgornjem diagramu obravnavamo položaj vozila. Vodoravna os je območje vrednosti za spremenljivko $ep(t)$. Glede na to, kakšen je $ep(t)$, sklepamo, kakšen je položaj vozila. Vozilo je lahko desno ali levo od trajektorije, blizu ali daleč. Vse to povemo z $ep(t)$, ki ima neko vrednost in predznak. Vendar, kakor smo že rekli, nam številke včasih niso všeč. Zato se odločimo, da bomo ločili samo pet različnih položajev, položaj zelo levo (*pzl*), položaj levo (*pl*), položaj pravilno (*pp*), položaj desno (*pd*) in položaj zelo desno (*pzd*).

V danem primeru bomo postavili meje oz. srednje vrednosti intervalov takole: vozilo je v pravilnem položaju, če je na trajektoriji, če je oddaljenost od trajektorije 0 m; vozilo je desno od trajektorije, če je od nje oddaljeno 0,1 m; vozilo je zelo desno, če je oddaljeno od trajektorije 0,2 m ali več. Za levo stran velja enak opis, le da so predznaki oddaljenosti negativni. Kaj pa vmesne vrednosti? Kaj bomo rekli, če je vozilo 0,05 m desno od trajektorije, ko je vmes med pravilnim položajem in položajem desno? Zamislimo si funkcije pripadnosti za različne položaje vozila. Za funkcije pripadnosti smo izbrali trikotnike, razen za skrajni vrednosti, za položaj zelo desno (*pzd*) in položaj

in our case, that is finding the actual position and orientation, and finding the corresponding membership grades. There are three diagrams on fig. 3. The upper is for position. The horizontal axis represents values of $ep(t)$. From $ep(t)$ we can conclude the position of the vehicle. The vehicle can be on the right or on the left side of the desired trajectory, near or far. All is said with $ep(t)$, which has a value and a sign. But, as the introductory statements said, we do not like numbers, so we decide to distinguish only five different positions (linguistic variables!), a very left position (*pzl*), left position (*pl*), correct position (*pp*) right position (*pd*) and very right position (*pzd*).

In our case we will set the limits or medium values of subintervals in this way: the vehicle is in the correct position if it is on the (desired) trajectory, and if the distance from the trajectory is 0 m; the vehicle is on the right side of the trajectory, if the distance is 0.1 m; the vehicle is very right if the distance is 0.2 m or more. We have negative signs for the left side. What about values in between? What shall we say if the vehicle is 0.05 m on the right side, when it is between the correct and right position? Imagine the membership function for position. We selected zelo levo (*pzl*). Če je $ep(t)$ > = *zd*, potem je funkcija pripadnosti *pzd* = 1. Podobno velja za zelo levi položaj. Sicer velja, da je funkcija pripadnosti za ustrezno srednjo vrednost enaka 1, levo in desno od srednje vrednosti pa linearno pada, za vrednosti *ep*, ki se razlikujejo za več ko za 0,1 m absolutno od srednje vrednosti, je enaka 0.

Na sredini diagrama je *ep* = 0, položaj vozila je pravilen, funkcija pripadnosti *pp* je 1. Središčne vrednosti posameznih intervalov oziroma funkcij pripadnosti so označene z *zl* (zelo levo), *l* (levo), *p* (pravilno, ni odstopanja), *d* (desno) in *zd* (zelo desno), torej je p = 0,0 m, d = 0,1 m, *zd* = 0,2 m, l = -0,1 m in *zl* = -0,2 m. Za vmesno vrednost torej, npr. $ep(t) = d/2$ = 0,05 m, sta različni od 0 samo funkciji pripadnosti *pp* in *pd*, obe sta enaki, *pp* = 0,5 in *pd* = 0,5. Vozilo je torej hkrati v pravilnem položaju in položaju desno. Ker imata funkciji pripadnosti enaki vrednosti in ker sta funkciji pripadnosti simetrični glede na trenutno vrednost *ep*, lahko sklepamo, da je vozilo prav v sredini med pravilnim položajem in položajem desno. Za $ep(t)$ = = $3d/4$ je *pp* = 0,25 in *pd* = *0,75;* vozilo je torej hkrati v pravilnem položaju in položaju desno, vendar bolj desno kakor v pravilnem položaju, kar se vidi iz tega, da je pripadnost *pd* večja od *pp*. Glede na narisane funkcije pripadnosti vidimo, da sta vedno samo dve različni od nič, razen če zavzame $ep(t)$ eno od središčnih vrednosti, tedaj je samo ena funkcija pripadnosti različna od nič in je enaka 1.

Če je $ep = -d/2$, sta *pp* in *pl* enaki 0,5, druge 0 (pripadnost je vedno pozitivna!).

Analogno velja za $es(t)$ in smer, razmere so prikazane na drugem diagramu na sliki 3. Za dani primer velja, da je *zd* = 0,25 radiana, *d* je polovica tega, torej *d* = 0,125 radiana. Definiranih je 5 funkcij pripadnosti za smer, in sicer smer zelo levo (*szl*), smer levo (*sl*), smer pravilna (*sp*), smer desno (*sd*), smer zelo desno (*szd*).

Bistvo prve faze je torej ugotoviti stanje sistema in ga diskretizirati s funkcijami pripadnosti, tako da samo ugotovimo, na katerem intervalu je posamezna spremenljivka. Za naš problem, ko merimo dve spremenljivki in imamo za vsako 5 funkcij pripadnosti, imamo 25 različnih kombinacij. Da določimo, na katerem podintervalu je posamezna spremenljivka, moramo določeno informacijo opustiti (prej smo v primeru 10-bitnega pretvornika predpostavili, da odrežemo spodnjih 8 bitov in ohranimo samo 2). Vendar ta informacija ni izgubljena, ohranjena je v vrednostih funkcij pripadnosti. Določeno informacijo smo opustili samo zato, da smo dobili končno, omejeno število mogočih stanj, tako da se lahko za vsako od teh stanj oziroma kombinacij odločimo, kako ukrepati. Delno pri drugi fazi, delno pri zadnji fazi pa potem to trenutno opuščeno informacijo zopet uporabimo.

triangles, except for the extreme positions, for the very left (*pzl*) and the very right position (*pzd*). If $ep(t)$ > = *zd*, then the membership function gives a value *pzd* = 1. Similar conclusions hold for the very left position. Otherwise, the membership functions give a value of 1 for the medium values of the argument, for values left or right from the medium values, function values go linearly to 0 and reach 0 when 0.1 m off (absolutely) from the medium values.

In the middle of the diagram is *ep* = 0, the position of the vehicle is correct, the value of the membership function *pp* is 1. The medium values of intervals (membership functions respectively) are denoted with *zl* (very left), *l* (left), *p* (correct, no deviation), *d* (right) and *zd* (very right), or in other words p = 0.0 m, d = 0,1 m, *zd* = 0.2 m, l = -0.1 m, and *zl* = -0.2 m. For an intermediate value of *ep*, for example $ep(t) = d/2$ = 0.05 m, only membership functions *pp* and *pd* are different from 0. Both are equal, *pp* = 0.5 and *pd* = 0.5. The vehicle is in the correct position and in the right position at the same time. Since both membership functions have the same value and as the membership functions are symmetric with respect to the actual value of *ep*, we can deduce that the vehicle is just in the middle between the correct position and the right position. For $ep(t)$ = = $3d/4$ is *pp* = 0.25 and *pd* = 0.75; the vehicle is both in the correct position and in the right position, but it is more in the right position than in the correct one, which follows from *pd* being greater than *pp*. We see from membership functions drawn that only two of them are different from 0 at the same time, except in the case when $ep(t)$ is equal to one of the middle values of a particular membership function, when only this function is different from 0 and equl to 1.

If $ep = -d/z$, then *pp* and *pl* are equal to 0.5, and all other membership function are 0 (values of membership functions are always positive!).

The same holds for $es(t)$ and the orientation, shown on the second diagram on fig. 3. In our case, *zd* = 0.25 radians, *d* is half of that, *d* = 0.125 radians. Five membership functions for orientation are defined, for orientation very left (*szl*), orientation left (*sl*), correct orientation (*sp*) orientation right (*sd*) and orientation very right (*szd*).

The essence of the first phase is finding out the state of the system and discretizing it with the help of membership functions so that only the interval on which a particular variable lies, is determined. For our problem, where two variables are measured, each with five membership functions, there are 25 combinations. To determine on which subinterval a particular variable is, we have to leave out some information (formerly with a 10-bit converter we assumed we cut off the lower 8 bits, and keep 2 only). However, this information is not lost, it remains in the values of the membership functions. A certain part of the information has been discarded only to get a finite, limited number of states, so that we can

Za primer postavimo, da je položaj vozila 0,05 m na desni ($ep(t)$ = 0,05 m) in da je smer vozila glede na trajektorijo $es(t)$ = – 0,025 radiana. Ugotovimo, da sta za ta položaj samo funkciji pripadnosti *pp* in *pd* različni od 0, in sicer sta *pp* = 0,5 in *pd* = 0,5. Pri smeri sta funkciji pripadnosti *sp* in *sl* različni od 0 in sta *sp* = 0,8 in *sl* = 0,2. Sklepamo, da je položaj po eni strani pravilen, po drugi desno, smer pa tudi po eni strani pravilna, po drugi pa levo. S to informacijo oziroma vrednostmi funkcij pripadnosti (*pp* = 0,5, *pd* = 0,5, *sp* = 0,8, *sl* = 0,2) gremo v drugo fazo.

2.2.2 *Druga faza*

V tej fazi na podlagi nabora pravil in informacij iz prve faze določimo, kako ukrepati. Pravila so pogojni (IF) stavki, kakršne poznamo iz programskih jezikov. Npr.: če je položaj tak in smer taka, potem naj bo krmilni signal tak. Enako razmišljanje, kakor pri vstopnih signalih oz. stanju uporabimo tudi za krmilni signal. Prav tako si ne želimo imeti opravka z veliko množico njegovih vrednosti, ker je v takšnem primeru izbira težka. Možnosti naj bo raje malo, da se lahko odločimo. Tako kakor pri *ep* in *es* izberemo tudi tu 5 (lahko bi jih več ali manj) različnih vrednosti za lego krmila (zasuka prednjega kolesa $u(t)$), in sicer teh pet vrednosti poimenujemo krmilo zelo levo (*kzl*), krmilo levo (*kl*), krmilo naravnost (*kn*), krmilo desno (*kd*) in krmilo zelo desno (*kzd*). V našem primeru velja, da so središčne vrednosti posameznih leg krmila *zl* = –0,8 radiana (pri *kzl* = 1), *l* = –0,4 radiana (pri *kl* = 1), *n* = 0,0 radianov (pri *kn* = 1), *d* = 0,4 radiana (pri *kd* = 1) in *zd* = 0,8 radiana (pri *kzd* = 1). Razmere so ponazorjene s tretjim diagramom na sliki 3.

Potem si izmislimo pravila. Npr.: če je položaj pravilen in smer pravilna, naj bo krmilo v običajni legi oziroma legi naravnost. Ali npr.: če je položaj desno in smer levo, naj bo krmilo v običajni legi. Ali: če je položaj desno in smer desno, naj bo krmilo zelo levo. In tako naprej za vse mogoče kombinacije vstopnih signalov (»obdelanih« s funkcijami pripadnosti). Najbolje je to sistematično urediti, da česa ne spregledamo. V našem primeru imamo pet različnih položajev in pet različnih smeri, zato sestavimo odločilno preglednico s 25 lokacijami:

decide what to do for each of the states. Later, partly in the second and partly in the third phase, we make use of this temporarily discarded information again.

As an example, assume the position of the vehicle to be 0.05 m to the right ($ep(t)$ = 0.05 m) and that the orientation of the vehicle with respect to the trajectory is $es(t)$ = –0.025 radians. We find for this position only membership functions *pp* and *pd* different from 0, they are *pp* = 0.5 and *pd* = 0.5. For orientation only functions *sp* and *sl* are different from 0 and are *sp* = 0.8 and *sl* = 0.2. We conclude that the position is correct and right, the orientation correct and left at the same time. With these values of membership functions (*pp* = = 0.5, *pd* = 0.5, *sp* = 0.8, *sl* = 0.2) we enter the second phase.

2.2.2 *Second phase*

In this phase, we decide from a collection of rules and from the information obtained in the first phase what should be done. The rules are IF statements, as known from programming languages. For example, if the position is such and orientation is such, then the control has to be such. The same kind of thinking as with input signals or states is valid for control signals. We want a small number of options to simplify decisions. As with *ep* and *es* we select five (it could be more or less) different values for the position of the steering wheel $u(t)$, we call these 5 values steering wheel very left (*kzl*), steering wheel left (*kl*), steering wheel straight ahead or normal position (*kn*), steering wheel right (*kd*) and steering wheel very right (*kzd*). In our case the numerical values for these steering wheel positions are *zl* = –0.8 rad (at *kzl* = 1), *l* = –0.4 rad (at *kl* = 1), *n* = 0.0 rad (at *kn* = 1), *d* = 0.4 rad (at *kd* = 1) and *zd* = 0.8 rad (at *kzd* = 1). The relations are illustrated in fig. 3, bottom diagram.

We must then invent the rules. For example: if the position is correct and the orientation is correct, then the steering wheel should be in the straight ahead position. Or: if the position is right and the orientation left, the steering wheel should be in the straight ahead position. Or: if the position is right and the orientation is right, the steering wheel should be in the very left position; and so forth for all possible combinations of input signals (after making use of the membership functions). We have five possible positions and five possible orientations in our case, so we make a decision table with 25 places:

	PZL	PL	PP	PD	PZD
SZL	KZD	KZD	KZD	KD	KN
SL	KZD	KZD	KD	KN	KL
SP	KZD	KD	KN	KL	KZL
SD	KD	KN	KL	KZL	KZL
SZD	KN	KL	KZL	KZL	KZL

V prvi vrstici preglednice so označbe posameznih položajev, kolikor jih ločimo, v prvem stolpcu pa označbe smeri. Znotraj preglednice najdemo ustrezno odločitev. Seveda je taka preglednica mogoča le v primeru, če so pogoji preprosti, v našem primeru obravnavamo hkrati vedno le dve spremenljivki, položaj in smer; v splošnem so lahko pravila bolj zapletena in jih ne moremo prikazati tako pregledno s preglednico. Namesto preglednice bi lahko zapisali 25 pravil:

1. če *pzl* in *szl* potem *kzd*,
2. če *pl* in *szl* potem *kzd*,
3. če *pp* in *szl* potem *kzd*,
4. če *pd* in *szl* potem *kd*,
5. če *pzd* in *szl* potem *kn*,

in tako naprej še za druge štiri vrstice.

Vidimo, da se pojavlja v zgornjih pravilih logična operacija IN. Uveljavimo jo kot konjunkcijo ali kako drugače. Običajno uporabljamo za izvedbo operacije IN minimalni operator, ker je preprost. Ta ustreza operatorju konjunkcije, če ga izvedemo npr. z diodnimi vrati. Skratka, najpogosteje izvedemo operacijo IN takole:

a IN b = minimum (a, b).

Zgornjih 5 pravil lahko potem napišemo takole:

1. *kzd* = minimum (*pzl*, *szl*),
2. *kzd* = minimum (*pl*, *szl*),
3. *kzd* = minimum (*pp*, *szl*),
4. *kd* = minimum (*pd*, *szl*),
5. *kn* = minimum (*pzd*, *szl*).

Za dani primer in predpostavljene vrednosti $ep(t)$ = 0,05 m in $es(t)$ = -0,025 rad, smo dobili *pp* = 0,5, *pd* = 0,5, *sp* = 0,8 in *sl* = 0,2.

Vsakemu položaju pripada stolpec preglednice, vsaki smeri vrstica v preglednici. Ker imamo dve funkciji pripadnosti položaja različni od 0 in dve funkciji pripadnosti smeri različni od 0, so s tem definirani štirje kvadratki v preglednici, ki ustrezajo štirim pravilom, ki jih moramo upoštevati:

1. če (*pp* in *sl*), potem *kd* oziroma *kd* = minimum (*pp*, *sl*),
2. če (*pd* in *sl*), potem *kn* oziroma *kn* = minimum (*pd*, *sl*),
3. če (*pp* in *sp*), potem *kn* oziroma *kn* = minimum (*pp*, *sp*),
4. če (*pd* in *sp*), potem *kl* oziroma *kl* = minimum (*pd*, *sp*).

Velja poudariti, da so *kd*, *kn*, in *kl* (ter *kzl* in *kzd*, ki jih ni v teh pravilih) funkcije pripadnosti, z vrednostmi med 0 in 1.

Govorimo o pravilih, videti pa so kot zmešnjava. Prvo pravilo zahteva, da krmilo obrnemo desno, četrto pa, da ga obrnemo levo. Kaj napraviti? Morda je najbolje pustiti vse skupaj pri miru. In to se tudi zares lahko zgodi, da se nič ne zgodi in vozilo pelje naprej kar z enako usmerjenosto krmila. Ta problem bomo rešili v tretji fazi. Trenutno se

In the first row of the table are labels for particular positions and labels for orientations in the first column. We read the table thus: if the position is very left (*pzl*) and the orientation is very left (*szl*), then turn the steering wheel to the very right (*kzd*). Or: if the position is correct (*pp*) and the orientation is right (*sd*) turn the steering wheel to the left. Such a table is, of course, possible only in simple cases. In our case, we have only two variables, but in general the rules are much more complicated and cannot be shown so clearly with a table. Instead of a table we could have written 25 rules:

1. if *pzl* and *szl* then *kzd*
2. if *pl* and *szl* then *kzd*
3. if *pp* and *szl* then *kzd*
4. if *pd* and *szl* then *kd*
5. if *pzd* and *szl* then *kn*

and so forth for the remaining four rows.

We see the operator AND in the conditions. This is achieved as a conjuction or in some other way. Usually, the operator minimum is employed to achieve AND, which corresponds to a conjunction realized with a diode gate. In short, the operator AND is most often realized in the following way:

a AND b = minimum (a, b).

Then the above 5 rules can be rewritten:

1. *kzd* = minimum (*pzl*, *szl*)
2. *kzd* = minimum (*pl*, *szl*)
3. *kzd* = minimum (*pp*, *szl*)
4. *kd* = minimum (*pd*, *szl*)
5. *kn* = minimum (*pzd*, *szl*).

For the assumed values of $ep(t)$ = 0.05 m and $es(t)$ = -0.025 rad in our case we obtained *pp* = 0.5, *pd* = 0.5, *sp* = 0.8 and *sl* = 0.2. Each particular position has a corresponding column in the table, each orientation a row. As we have 2 membership function for position different from 0 and 2 membership functions for orientation different from 0, these functions define two rows and two columns in the table, which correspond to 4 places in the table which again correspond to 4 rules which have to be considered. The rules which have to be obeyed are:

1. if (*pp* and *sl*) then *kd* or *kd* = minimum (*pp*, *sl*)
2. if (*pd* and *sl*) then *kn* or *kn* = minimum (*pd*, *sl*).
3. if (*pp* and *sp*) then *kn* or *kn* = minimum (*pp*, *sp*)
4. if (*pd* and *sp*) then *kl* or *kl* = minimum (*pd*, *sp*)

Let us stress that *kd*, *kn* and *kl* (also *kzl* and *kzd*, which are not present in these rules) are values of membership function and so in the range between 0 and 1.

We talk about rules but it looks like a mess. The first rule requires us to turn the wheel to the right, the fourth rule to turn it to the left. What to do? Probably the best thing is to let the vehicle and the steering wheel take their course. Indeed, it can often occur, that nothing happens, that the vehicle keeps the old direction.

pomudimo pri drugem in tretjem pravilu, obe zahtevata, da je krmilo usmerjeno naravnost oziroma, da je v običajnem položaju. Glede na to, kako *kn* določamo, je vrednost *kn*, določena po drugem pravilu, lahko različna od vrednosti *kn*, določene po tretjem pravilu. Odločimo se, da izberemo izmed njiju maksimalno vrednost, kar ustreza operatorju disjunkcije, če ga realiziramo z diodnimi vrati.

Če je v nekem trenutku zahtevan isti položaj krmila z več pravili, potem za končno vrednost funkcije pripadnosti tega položaja izberemo maksimalno vrednost (ali ukrepamo kako drugače, izbira maksimuma je samo ena od možnosti, preprosta za izvedbo). Torej:

kn = maksimum (*kn* (po 2. pravilu), *kn* (po 3. pravilu)).

Če upoštevamo pravila in določimo *kl*, *kn* in *kd* z uporabo minimuma in maksimuma, dobimo kot rezultat naslednje vrednosti funkcij pripadnosti za krmilno spremenljivko $u(t)$:

kl = 0,5
kn = maks (0,2, 0,5) = 0,5
kd = 0,2

2.2.3 *Tretja faza*

Ostala je še zadnja, tretja faza. V drugi fazi smo se odločili, kaj bomo storili. Izbrani ukrepi si lahko med seboj nasprotujejo. Problem istovrstnih zahtev, a z različno težo, smo že rešili, ostane še uskladitev različnih zahtev. Krmilo je namreč lahko le v enem položaju, ne more biti v dveh ali treh ali še več hkrati. Če smo na primer kot rezultat upoštevanja pravil dobili zahtevi *kd* = = 1 in *kl* = 1, je očitno, da obe zahtevi hkrati najbolje upoštevamo tako, da damo krmilo v normalni položaj, natančno med oba zahtevana položaja. Lahko rečemo, da smo naredili povprečje obeh zahtev. In to tudi delamo. Če imamo zahtevi npr. *kn* = 0,5 in *kd* = 0,5, je ustrezni položaj krmila $u(t) = d/2$, na sredini med normalnim položajem in položajem desno. Kako pa, če funkciji pripadnosti nista enaki? Če je npr. *kn* = 0,6 in *kd* = 0,3. Tako kakor prej naredimo povprečje. Prej je bilo to preprosto, sedaj je nekoliko bolj zapleteno. Preprosta pot do določanja povprečnih vrednosti je določitev skupnega težišča površin, ki jih definirajo funkcije pripadnosti za $u(t)$. Na sliki 4 vidimo, da vrednosti *kn* = 0,6 priredimo trapez, simetrično narisan glede na središčno vrednost *n*. Narisan je z debelejšo črto. Podobno lahko vrednosti *kd* = 0,3 priredimo trapez, narisan simetrično na središčno vrednost *d*. Trapeza se ujemata po obliki s funkcijama pripadnosti, njuna višina pa je kar enaka ustrezni funkciji pripadnosti. Osnovna geometrija zadošča, da najdemo skupno težišče obeh likov, težišče je v točki u = 0,377 *d*.

This problem will be solved in the third phase. Now we will take a closer look at the second and the third rule. Both require the steering wheel to be in the normal position, straight ahead. Following the described procedures for determining *kn*, it can happen that the value of *kn*, as determined by the second rule is different from *kn* as determined by the third rule. We choose the maximum value of both, which corresponds to the disjunction operator, if realized with a diode gate.

If several rules require the same position of the steering wheel at a certain moment, then for the final value of the membership function for this particular steering wheel position the maximum value is chosen (or some other value, the maximum is only one of the possibilities, simple to realize). That is:

kn = maximum (*kn* (obtained by rule 2), *kn* (obtained by rule 3))

If rules are taken care of and we determine *kl*, *kn* and *kd* with minimum and maximum operators, we finally get the following values for membership functions for the control variable $u(t)$ (steering weel position):

kl = 0.5
kn = max (0.2, 0.5) = 0.5
kd = 0.2.

2.2.3 *Third phase*

The last, third phase, remains. In the second phase, we decided what to do. But the decisions made can oppose each other. The problem of the same requirements but with different weights has already been solved, what remains are different requirements. The steering wheel can be in one position only, not in two, three or even more positions at the same time. If with the rules we got the requirements *kd* = 1 and *kl* = 1 it is clear that both requirements are best fulfilled in such a way that we put the steering wheel in the normal position, halfway between the two required positions. We can say that we averaged the two requirements. If we have requirements *kn* = 0.5 and *kd* = = 0.5, then the position of the steering wheel is $u(t) = d/2$, halfway between the normal position and the right position. But what then, when the membership functions are not equal, have different values, if say *kn* = 0.6 and *kd* = 0.3? As before we average. To get the average before was a trivial procedure, now it is a little more complicated. An easy way to average is to determine the centroid or common »center of gravity« of the surfaces, defined by membership functions for $u(t)$. On figure 4, we see, how a trapezoid is determined to a value of *kn* = 0.6, drawn symmetrically with regard to the value *n*. It is drawn with a thicker line. Similarly can we arrange to the value *kd* = 0.3 a trapezoid, draw symetrically with regard to the value *d*. Both trapezoids conform to the membership functions, their heights are simply the corresponding values of the membership functions. Basic geometry suffices to find the common center of gravity of the two figures or surfaces, the centroid is in the point u = 0,377 *d*.

Za našo nalogo, ko smo določili $kl = 0,5$, $kn = 0,5$ in $kd = 0,2$, imamo razmere, kakršne so narisane na drugem diagramu slike 4. Abscisa težišča debelo izvlečenih likov je v točki $u = -0,21\ d$. V obeh primerih sta točki določeni z navpično črto.

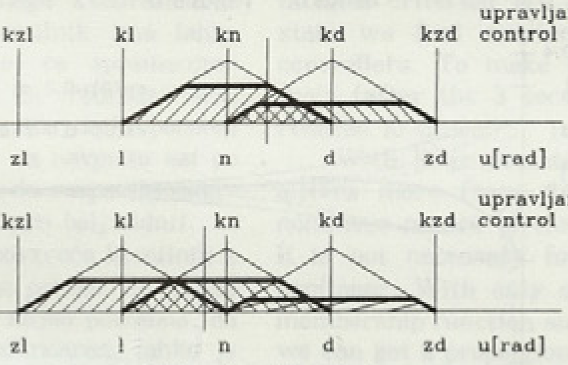
S tem je tretja faza sklenjena. Potem se vse ponovi: meritev stanja, določitev funkcij pripadnosti za merjene spremenljivke, upoštevanje pravil, določitev krmilne spremenljivke itn.

Vidimo, da informacijo, kakšno je dejansko stanje sistema, skušamo uporabiti v celoti, saj lahko na določevanje $u(t)$ gledamo kot na neke vrste interpolacijo. S funkcijami pripadnosti samo za krajši čas, da se laže odločimo, napravimo sliko bolj grobo, potem, po odločitvi, pa jo zopet skušamo upoštevati v vseh podrobnostih.

In our case, with $kl = 0.5$, $kn = 0.5$ and $kd = 0.2$, we have a situation as presented on the second chart of Figure 4. The abscissa of the center of gravity or centroid of the shapes drawn in thick lines is at the point $u = -0.21\ d$. In both cases, the points are marked with a vertical line.

This is the end of the third phase. Then everything is repeated, the measurement of the state, the values for membership functions of the measured variables are calculated, then the rules, then the control variables, and so on.

We try to use the whole information on the state of the system, and we can view the determination of $u(t)$ as some kind of interpolation. With the membership functions, some bits of information were left out only temporarily to get a coarser picture. Now, in the third phase, after the rules, we try to use them again.



Slika 4 — Figure 4

2.3 Primerjava in razlaga

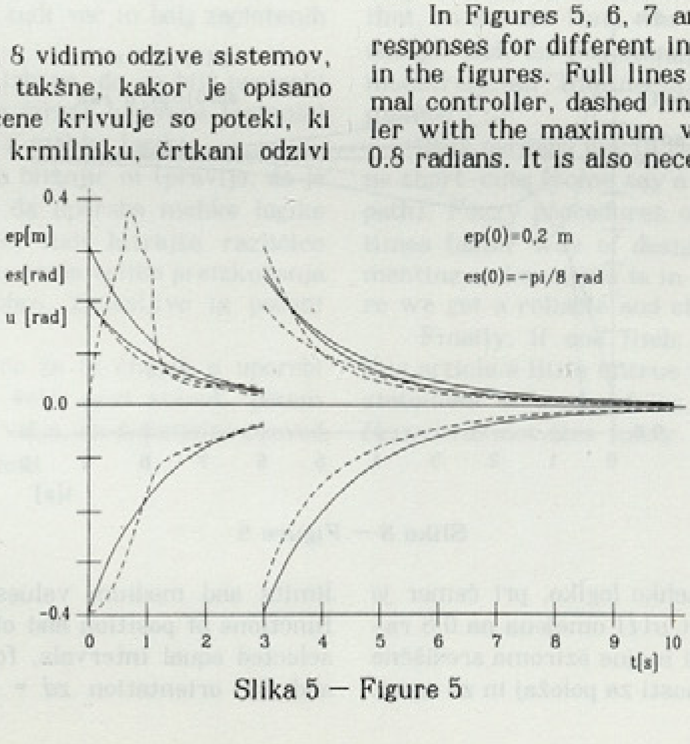
Da bi ocenili kakovost krmilnika, določenega z uporabo mehke logike, smo simulirali obnašanje vozila in krmilnega sistema, v enakih razmerah kakor v [8].

Na slikah 5, 6, 7 in 8 vidimo odzive sistemov, če so začetne razmere takšne, kakor je opisano na slikah. Polno izvlečene krivulje so poteki, ki ustrezajo optimalnemu krmilniku, črtkani odzivi

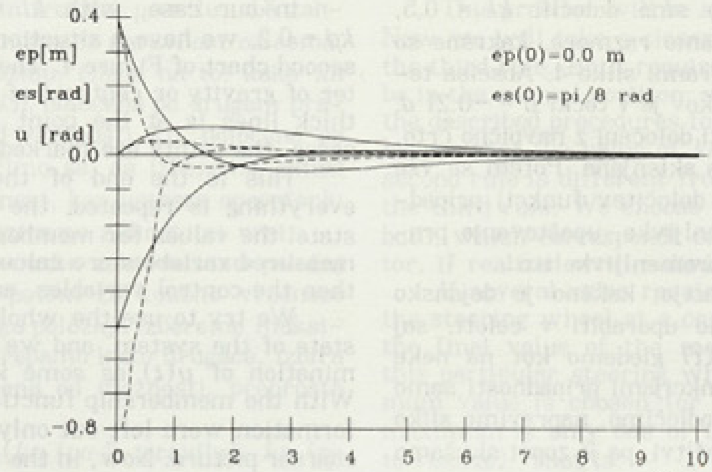
2.3 Comparison and comments

To estimate the quality of the controller as determined with the fuzzy method, we simulated the behavior of the vehicle and the controller under the same conditions as in [8].

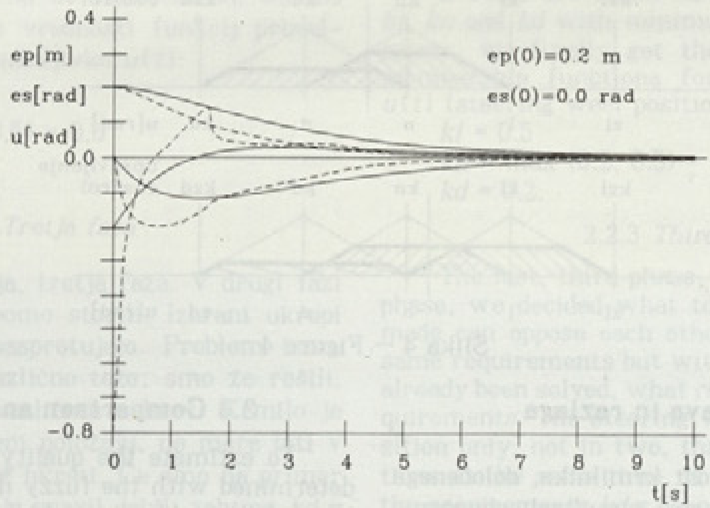
In Figures 5, 6, 7 and 8, we have systems responses for different initial conditions as given in the figures. Full lines correspond to the optimal controller, dashed lines to the fuzzy controller with the maximum value of $u(t)$ limited to 0.8 radians. It is also necessary to select interval



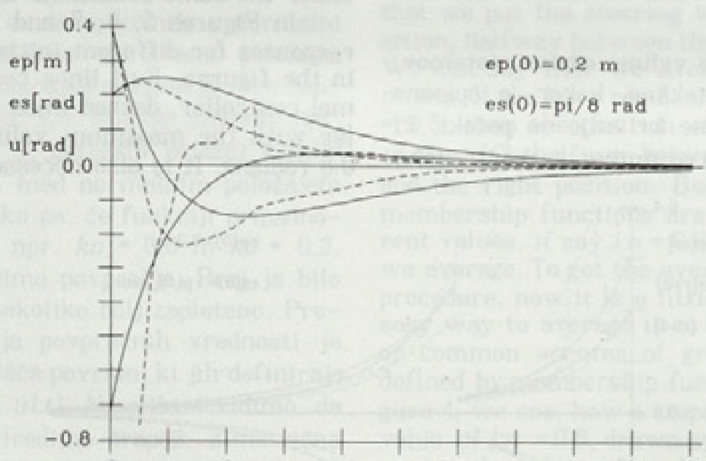
Slika 5 — Figure 5



Slika 6 — Figure 6



Slika 7 — Figure 7



Slika 8 — Figure 8

ustrezajo krmilniku z mehko logiko, pri čemer je bila maksimalna vrednost $u(t)$ omejena na 0,8 radiana. Izbrati je treba tudi mejne oziroma središčne vrednosti funkcij pripadnosti za položaj in za smer.

limits and medium values for the membership functions of position and of orientation. We have selected equal intervals, for position zd = 0.2 m and for orientation zd = 0.25 rad. With these

Sprejeta je delitev na enake intervale, za položaj velja, da je $zd = 0{,}2$ m, za smer pa je $zd = 0{,}25$ radiana. S temi vrednostmi so potem določene tudi druge ($d = zd/2$, $l = -d$, $zl = -zd$), če bi pa bili intervali neenaki, bi morali definirati več vrednosti.

Na slikah so po tri krivulje, $ep(t)$, $es(t)$ in $u(t)$. Katera krivulja ustreza kakemu signalu, lahko ugotovimo iz ustreznih začetnih pogojev, tista krivulja, ki ne začne na nobenem od začetnih pogojev za $ep(t)$ in $es(t)$ ustreza poteku $u(t)$.

Kaj ugotovimo? Tudi krmilnik z mehko logiko se izkaže kot primerna rešitev. Če spreminjamo *zd* za položaj in smer, lahko dobimo tudi zelo hitre odzive. Parametri krmilnika z mehko logiko so bili tako izbrani, da so hitrosti odzivov približno primerljive z odzivom optimalnega krmilnika, dobljenega po postopku linearnega kvadratičnega upravljanja. Tudi optimalni krmilnik ima lahko drugačne, tudi hitrejše poteke, če spremenimo uteži v integralskem kriteriju. Za vrednosti okoli ravnotežnega stanja vidimo, da so poteki podobni (zato je tudi na sliki 5 merilo na navpični osi po 3 sekundah 10-krat zmanjšano, da so poteki odzivov v bližini ravnotežnih vrednosti bolj vidni).

Pri večjih odstopanjih pa povzroča krmilnik z mehko logiko nekoliko drugačne poteke, do izraza pride njegova nelinearnost. Ni nujno potrebno, da je krmilnik z mehko logiko nelinearen, lahko je linearen. Če bi imeli eno samo področje in eno samo funkcijo pripadnosti in ne 5, tako kakor smo storili, bi dobili v bistvu proporcionalni krmilnik, kar bi še poenostavilo izvedbo in odzivi bi bili bolj gladki (tudi $u(t)$).

Krmilniki z mehko logiko so primerni predvsem pri nelinearnostih, pri zapletenih objektih. Tam je treba predpisati tudi več in bolj zapletenih pravil.

Primer je bil tako izbran, da so bili postopki preprosti in so prišle do izraza bistvene lastnosti načrtovanja krmilnikov z mehko logiko.

Pripomniti velja, da bližnjic ni (pravijo, da je bližnjica najdaljša pot), da uporaba mehke logike sicer omogoča drugačno, tudi hitrejšo različico načrtovanja, da pa vseeno terja veliko preizkušanja in analiz, da dobimo dobro, zanesljivo in poceni rešitev.

In čisto za konec, če za ta članek o uporabi mehke logike ravno ne velja prvi stavek, potem upam, da tudi četrti ne velja, da pisanje ni preveč megleno oziroma zabrisano.

values, other are determined ($d = zd/2$, $l = -d$, $l = -zd$). For unequal intervals, more values have to be defined.

Three curves are given in each figure, $ep(t)$, $es(t)$ and $u(t)$. Which is which can be found out from the initial conditions. The curve which does not correspond to either initial condition is $u(t)$.

What conclusion can we draw? The fuzzy controller is also a suitable solution for guiding a mobile robot. If *zd* for position and orientation is varied, very fast responses can be obtained also. Parameters of the fuzzy controller were chosen in such a way that the responses are comparable with the responses of the linear quadratic optimal controller. The linear quadratic optimal controller can also have faster responses, if weights in the integral criterion are changed. Near the steady state we find similar responses for the two controllers. To make this clearer, the vertical scale (after the 3 secs mark) on figure 5 was reduced 10 times.

With large deviations, the fuzzy controller differs more from the optimal controller, the nonlinear nature of the former is more evident. It is not necessary for the fuzzy control to be nonlinear. With only one interval and only one membership function and not 5 as in our problem, we can get a proportional controller (i.e. a linear one), which would have simplified the realization and the responses would have been smoother (also $u(t)$).

It is with nonlinear systems and complicated systems that fuzzy control comes into its own. There, many more rules are required.

However, the case example was so chosen that, on the one hand, the procedures were simple enough and, on the other, it was possible to demonstrate all the necessary steps of a fuzzy method.

It is perhaps worth remarking, that there are no short-cuts (some say a short-cut is the longest path). Fuzzy procedures offer a different, sometimes faster way of design, but a lot of experimenting and analysis is in any case required before we get a reliable and cheap solution.

Finally, if one finds the first statement in this article a little untrue then one hopes the forth statement to be also false, that this talking about fuzzy was not also fussy.

3. LITERATURA
3. REFERENCES

[1] Zadeh, L.: Fuzzy Sets. Information and Control. Vol. 8. 1965.

[2] Zadeh L.: Making Computers Think like People. IEEE Spectrum, Vol. 21. 4, Aug. 1984.

[3] Zadeh L.: Fuzzy Logic, Computer, vol. 21. 4, Apr. 1988.

[4] Maiers, J.–Sherif, Y.S.: Applications of Fuzzy Set Theory, IEEE Trans. on Systems, Man., and Cyber., Vol. SMC–15, 1 Jan/Feb 1985.

[5] Neuro–Fuzzylogik, Technische Rundschau, 8, 1992.

[6] Liang, G.S.–Wang, M.J.J.: Personnel Placement in a Fuzzy Environment, Computers Ops. Res., Vol. 19, 2, 1992.

[7] Russel, P.: Knjiga o možganih, DZS, Ljubljana, 1990.

[8] Hemami, A.–Mehrabi, M.G.–Cheng, R.M.H.: Synthesis of an Optimal Control Law for Path Tracking in Mobile Robots, Automatica, vol. 28, 2, Mar. 1992.

[9] Abel, D.: Fuzzy Control – eine Einführung ins Unscharfe, Automatisierunstechnik, Vol. 39, 12, 1991.

[10] Magazin für Computerpraxis, št. 3, 9, 10, 11, 12 (1991), 1 (1992).

Avtorjev naslov: prof. dr. Alojz Hussu, dipl. inž.
Fakulteta za strojništvo
Univerze v Ljubljani
Aškerčeva 6
Ljubljana, Slovenija

Author's Address: Prof. Dr. Alojz Hussu, Dipl. Ing.
Faculty of Mechanical Engineering
University of Ljubljana
Aškerčeva 6
Ljubljana, Slovenia

Prejeto:
Received: 12.6.1992

Sprejeto:
Accepted: 15.8.1992