



# Rýchlostné metódy reaktívnej navigácie mobilných robotov

Cieľom článku je opísať hlavné metódy rýchlostnej reaktívnej navigácie mobilných robotov. Článok sa zaoberá metódami dynamického okna, rýchlostného oblúka a Nearness diagramom. Všetky rýchlostné metódy sa odlišujú od klasických metód reaktívnej navigácie tým, že okrem výsledného predpísaného smeru definujú pre pohyb robota aj rýchlosť, akou sa má pohybovať. Tieto metódy teda zohľadňujú aj dynamické vlastnosti robota.

## Úvod [1]

Klasické metódy reaktívnej navigácie mobilných robotov predpisujú robotu len smer pohybu, prípadne istým spôsobom kvantifikovanú veľkosť pohybu týmto smerom. Rýchlostné metódy pracujú nad tzv. priestorom rýchlostí, t. j. dvojprvkovej množiny  $(\omega, v)$ , kde  $\omega$  predstavuje uhlovú rýchlosť robota vzhľadom na jeho ťažisko a  $v$  predstavuje rovnako sa vzťahujúcu translačnú rýchlosť robota. Výsledkom aplikácie týchto rýchlostných metód je potom optimálna dvojica  $(\omega, v)$ , ktorá predpisuje správanie robota v najbližšom kroku. Jednotlivé metódy sa líšia v spôsobe určenia tejto dvojice.

## 1. Metóda dynamického okna [1] [2]

Metóda reaktívnej navigácie, nazvaná aj metóda dynamického okna (z angl. Dynamic window approach) pracuje nad priestorom rýchlostí. Táto metóda volí v každom kroku podľa istých kritérií najlepšiu translačnú a rotačnú rýchlosť. Trajektória robota pozostáva z oblúkov. Samotnú metódu možno opísať v týchto krokoch:

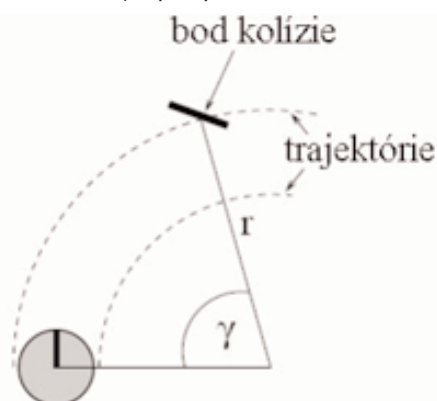
1. Vytvorenie množiny  $V_s$ , ktorá reprezentuje obmedzenia maximálnych rýchlostí. Množina  $V_s$  teda obsahuje také dvojice  $(\omega, v)$ , ktoré sú z pohľadu vlastností pohonov robota dosiahnuteľné.
2. Vytvorenie množiny  $V_a$  reprezentujúcej obmedzenia rýchlostí na základe informácií o prekážkach. Množina  $V_a$  obsahuje také dvojice  $(\omega, v)$ , pri ktorých je robot schopný zastaviť ešte pred prekážkou na príslušnom dráhovom oblúku so zakrivením  $c = \frac{\omega}{v}$ . Množina  $V_a$  je definovaná:

$$V_a = \left\{ (\omega, v) \mid v \leq \sqrt{2 \text{dist}(\omega, v) \dot{v}_b} \wedge \omega \leq \sqrt{2 \text{dist}(\omega, v) \dot{\omega}_b} \right\} \quad (1)$$

kde  $\dot{v}_b$  a  $\dot{\omega}_b$  sú zrýchlenia pri brzdení a funkcia  $\text{dist}(\omega, v)$  počíta príslušnú dĺžku oblúka (obr. 1):

$$\text{dist}(\omega, v) = r(\omega, v) \gamma(\omega, v) \quad (2)$$

kde  $r$  je polomer oblúka a  $\gamma$  je opísaný uhol.



Obr.1 Určenie dĺžky dráhy k prekážke

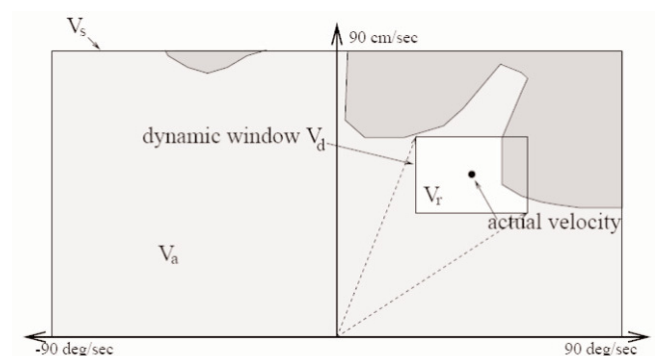
3. Vytvorenie množiny  $V_d$  (tzv. dynamického okna), ktorá reprezentuje obmedzenie vyplývajúce z dynamických vlastností robota (hodnoty zrýchlení  $\dot{v}$  a  $\dot{\omega}$ ), dosiahnutých počas krátko časového intervalu  $t$  (odvodený od dĺžky kroku algoritmu). Množina  $V_d$  je definovaná:

$$V_d = \left\{ (\omega, v) \mid v \in \langle v_a - \dot{v}t; v_a + \dot{v}t \rangle \wedge \omega \in \langle \omega_a - \dot{\omega}t; \omega_a + \dot{\omega}t \rangle \right\} \quad (3)$$

kde  $v_a$  a  $\omega_a$  sú aktuálne rýchlosti robota.

Prieknikom všetkých troch množín je množina  $V_r$  (obr. 2), na základe ktorej sa vyberá optimálna hodnota  $(\omega, v)$ :

$$V_r = V_s \cap V_a \cap V_d \quad (4)$$



Obr.2 Definícia množiny  $V_r$  [1] [2]

Pre každú dvojicu  $(\omega, v)$  je určená funkčná hodnota funkcie:

$$f(\omega, v) = a \cdot \text{heading}(\omega, v) + b \cdot \text{velocity}(\omega, v) + c \cdot \text{dist}(\omega, v) \quad (5)$$

kde  $\text{heading}(\omega, v)$  je miera postupu robota k cieľu (definovaná ako  $180 - \tau$ , kde  $\tau$  je odchýlka budúceho smeru robota od smeru k cieľu), je  $v$  najjednoduchšom prípade hodnota translačnej rýchlosti  $v$  a  $\text{dist}(\omega, v)$  je dĺžka oblúčovej dráhy k prekážke. Ak na danom oblúku

neleží žiadna prekážka, funkcia  $\text{dist}(\omega, v)$  vracia vysokú hodnotu. Jednotlivým častiam účelovej funkcie, ktoré sú normalizované na interval  $\langle 0, 1 \rangle$ , sa dá nastaviť veľkosť vplyvu pomocou parametrov  $a$ ,  $b$  a  $c$ . Obvykle bývajú tieto parametre nastavené tak, aby bol zachovaný rýchly pohyb robota s dodržaním veľkej vzdialenosti od prekážok a s rýchlym smerovaním k cieľu. Nový smer a rýchlosť pohybu sú stanovené na základe maximálnej hodnoty z funkčných hodnôt.



**2. Metóda rýchlostného oblúka [1] [3]**

Táto metóda patrí medzi metódy reaktívnej navigácie, ktoré berú do úvahy aj kinematické a dynamické vlastnosti. Podobne ako predošlá metóda pracuje nad priestorom rýchlostí. Medzi obmedzenia, ktoré táto metóda berie do úvahy, možno zaradiť rýchlosť a zrýchlenie robota, prekážky v priestore pretransformované do priestoru rýchlostí a obmedzenia od senzorov vzdialeností.

Zobrazenie prekážok v priestore rýchlostí sa vykonáva pomocou prídania tretieho rozmeru s hodnotami dĺžok oblúčkových dráh od robota k prekážke, ktoré zodpovedajú jednotlivým dvojiciam  $(\omega, v)$ . Ak berieme do úvahy viac prekážok, je zrejmé, že pre rovnaké dvojice  $(\omega, v)$  môžu existovať rôzne hodnoty dĺžky oblúčkovej dráhy. Preto sa k danej dvojici priraduje minimálna hodnota z možných dĺžok. Ak je dĺžka nejakej dráhy nekonečná, čiže robot by nenarazil s danou konfiguráciou  $(\omega, v)$  do žiadnej prekážky, hodnota dĺžky oblúčkovej dráhy je limitovaná na hodnotu  $L$ . Veľkosť  $L$  je odvodená od maximálneho meracieho rozsahu snímačov robota. Ak by však takáto dráhová funkcia vstupovala do optimalizačného procesu výberu dvojice  $(\omega, v)$ , bolo by to výpočtovo náročné, čo odporuje požiadavke nutnosti reaktívnej navigácie pracovať online. Preto sa pristupuje k istým zjednodušeniam. Prvé zjednodušenie je nahradenie prekážok prekážkami s kruhovým tvarom. Zobrazenie prekážok nad priestorom rýchlostí sa aproximuje diskretnými intervalmi s konštantnou hodnotou dĺžky dráhy. Kruhový tvar prekážok potom umožňuje výpočet minimálneho a maximálneho zakrivenia oblúčkových dráh dotýkajúcich sa prekážky (obr. 3):

$$c_{\min} = \frac{2(x_{obs} - y_{obs})}{\sqrt{x_{obs}^2 + y_{obs}^2 + r_{obs}^2}}$$

$$c_{\max} = \frac{2(x_{obs} + y_{obs})}{\sqrt{x_{obs}^2 + y_{obs}^2 + r_{obs}^2}}$$

(6)

kde  $x_{obs}$  a  $y_{obs}$  sú súradnice stredu prekážky a  $r_{obs}$  je jej polomer vrátane hodnoty, o ktorú sa zvyknú prekážky rozšíriť pre rozmery robota (robot potom môže byť pretransformovaný na jediný bod). Potom možno určiť aj body dotyku s prekážkou (obr. 3):

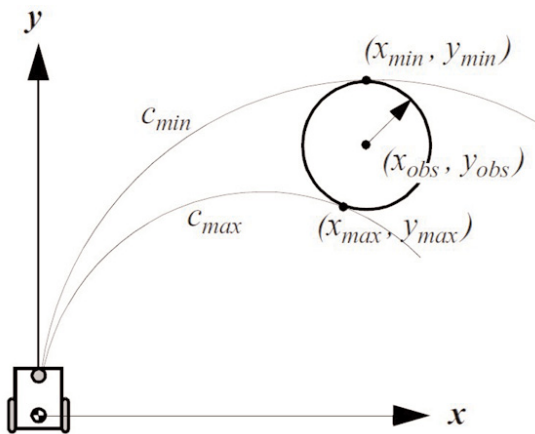
$$x_{\min} = \frac{(x_{obs} - r_{obs})}{(1 - c_{\min} r_{obs})}$$

$$y_{\min} = \frac{y_{obs}}{(1 - c_{\min} r_{obs})}$$

$$x_{\max} = \frac{(x_{obs} + r_{obs})}{(1 - c_{\max} r_{obs})}$$

$$y_{\max} = \frac{y_{obs}}{(1 - c_{\max} r_{obs})}$$

(7)



Obr.3 Dotýčnicové dráhové oblúky k prekážke [1]

Dĺžka dráhového oblúka k prekážke je potom definovaná ako:

$$d_c(c, obs) = \begin{cases} y_i & c = 0 \\ \frac{1}{c_i} \gamma & c \neq 0 \end{cases} \quad (8)$$

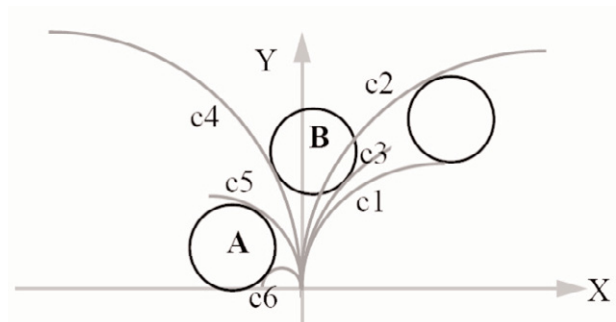
kde  $c_i$  je príslušné zakrivenie oblúka,  $y_i$  je súradnica bodu dotyku a  $\gamma$  je opísaný uhol oblúka v radiánoch a je určený ako:

$$\gamma = \begin{cases} \arctg \left( \frac{y_i}{x_i - \frac{1}{c_i}} \right) & c_i < 0 \\ \pi - \arctg \left( \frac{y_i}{x_i - \frac{1}{c_i}} \right) & c_i > 0 \end{cases} \quad (9)$$

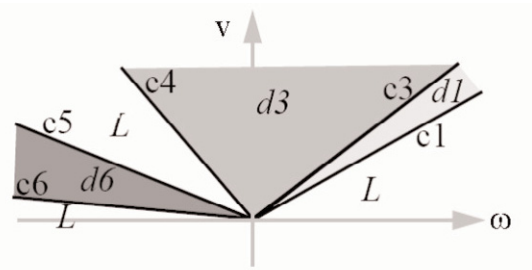
kde  $c_i$  a  $y_i$  sú súradnice bodu dotyku. Hodnota dráhovej funkcie ohraničená zakriveniami  $c_{\min}$  a  $c_{\max}$ , ktorým zodpovedajú príslušné hodnoty  $(\omega, v)$ , sa vypočíta zo vzťahu:

$$d_v(\omega, v, obs) = \begin{cases} \min[d_c(c_{\min}, obs), d_c(c_{\max}, obs)] & c_{\min} \leq \frac{\omega}{v} \leq c_{\max} \\ \infty & \text{inak} \end{cases} \quad (10)$$

Výpočet dráhovej funkcie treba aplikovať pre všetky prekážky. Ak sa niektoré intervaly prekrývajú, prieniku je priradená hodnota dráhy, ktorá je najnižšia.



Obr.4 Hraničné dráhové oblúky [3]



Obr.5 Intervaly zakrivenia v priestore rýchlostí [3]

Na priestor rýchlostí treba aplikovať takéto obmedzenia (obr. 5):

$$v \leq v_{\max}$$

$$v \geq 0$$

$$\omega \leq \omega_{\max}$$

$$\omega \geq -\omega_{\max}$$

$$d_v(\omega, v, obs) \leq L \quad (11)$$



Ďalej treba aplikovať obmedzenia vyplývajúce z dosiahnuteľnosti v nasledujúcom kroku:

$$\begin{aligned} \omega &\geq [\omega_a - (\varepsilon_{\max} T)] \\ \omega &\leq [\omega_a + (\varepsilon_{\max} T)] \\ v &\leq [v_a + (a_{\max} T)] \end{aligned} \quad , (12)$$

kde  $\omega_a$  a  $v_a$  predstavujú okamžité rýchlosti,  $\varepsilon_{\max}$  maximálne uhlové zrýchlenie,  $a_{\max}$  maximálne translačné zrýchlenie a  $T$  je perióda kroku algoritmu. Ak sú splnené všetky obmedzenia, potom možno vybrať vhodnú dvojicu  $(\omega, v)$  maximalizáciou funkčnej hodnoty účelovej funkcie:

$$f(\omega, v) = \alpha_1 \text{speed}(v) + \alpha_2 \text{dist}(\omega, v) + \alpha_3 \text{head}(\omega)$$

$$\begin{aligned} \text{speed}(v) &= \frac{v}{v_{\max}} \\ \text{dist}(\omega, v) &= \frac{d_L(\omega, v, \text{obs})}{L} \\ \text{head}(\omega) &= \frac{1 - |\tau - \omega T_c|}{\pi} \end{aligned} \quad (13)$$

Sčítance sú normalizované do intervalu  $\langle 0, 1 \rangle$ ,  $d_L$  predstavuje funkciu  $d_V$  zohľadňujúcu obmedzenie  $L$ ,  $\tau$  je smer k cieľu a  $T_c$  je zvolená časová konštanta. Parametrami  $\alpha$  možno ovplyvniť charakter správania robota, pričom prvý sčítanec je zodpovedný za rýchly chod robota, druhý za preferovanie veľkých dĺžok dráh k stretu s prekážkou a tretí za minimalizovanie rozdielu medzi smerovaním robota a smerom k cieľu.

Metóda rýchlostného oblúka pri aplikácii bez globálneho navigačného systému zlyháva v prípadoch existencie lokálnych miním v prostredí. Nevýhodou tejto metódy je aj použitie kruhového tvaru reprezentácie prekážok, čo v niektorých prostrediach môže spôsobiť veľké problémy. Takýmto prostredím môže byť dlhý koridor. Z tohto dôvodu bola metóda rýchlostného oblúka rozšírená na metódu pásového oblúka. Táto metóda počíta požadované pásy v prostredí, porovnáva teda dĺžku a šírku pásu s najbližšou prekážkou. Pás s najlepšimi vlastnosťami je určený podľa optimalizačnej funkcie. Ak sa robot nenachádza v tomto pásu, potom je jeho natočenie zmenené tak, aby sa do tohto pásu dostal. Táto metóda dosahuje lepšie výsledky ako pôvodná metóda rýchlostného oblúka. Má však jednu nevýhodu v podobe veľkej citlivosti výberu parametrov v optimalizačnej funkcii na správanie robota.

### 3. Nearness diagram [1] [5] [4] [6]

Táto metóda reaktívnej navigácie využíva dva typy diagramov. Prvý sa nazýva PND (Nearness diagram zo stredu). PND identifikuje regióny a vyberá jeden z nich pre zvolený pohyb robota. Druhý sa nazýva RND (Nearness diagram od robota). RND ohodnocuje bezpečnosť robota pri pohybe. Tieto informácie sú potom využité na identifikáciu piatich všeobecných situácií, ktoré môžu nastať.

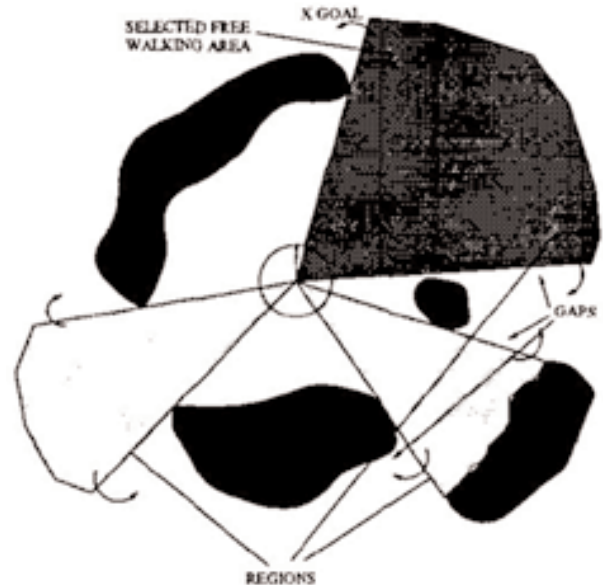
Metodológia Nearness diagram používa sektorovú reprezentáciu prostredia. Definujme bod  $c$ , ktorý je ťažiskom robota a zároveň stredom sektorovej reprezentácie prostredia. Nech  $n$  je počet sektorov a  $\theta$  je natočenie robota. Nech  $\delta$  je funkcia, ktorá pri známej konfigurácii robota  $q$  a bodu  $c$  udáva vektor  $\delta_i(q, c)$  tak, že je to najmenšia vzdialenosť v sektore  $i$ , pričom  $\theta$  sa nachádza v strede sektora  $\frac{n}{2}$ . Potom je diagram PND definovaný ako (obr. 9):

$$\begin{aligned} D_i = PND_i(q, c) &= d_{\max} + l - \delta_i(q, c) \quad \delta_i(q, c) > 0 \\ D_i = PND_i(q, c) &= 0 \quad \text{inak} \end{aligned} \quad (14)$$

kde  $d_{\max}$  je maximum funkcie  $\delta$  (teda použitý maximálny merací rozsah) a  $l$  je maximálna vzdialenosť medzi dvoma bodmi robota (pri kruhovom tvare robota je to priemer robota). RND je definovaný (obr. 10):

$$\begin{aligned} D_i = RND_i(q, c) &= d_{\max} + E_i(c) - \delta_i(q, c) \quad \delta_i(q, c) > 0 \\ D_i = RND_i(q, c) &= 0 \quad \text{inak} \end{aligned} \quad (15)$$

kde  $E_i(c)$  je funkcia závislá od geometrie robota. Pre robot kruhového tvaru  $E_i(c)$  zodpovedá polomeru robota. Vo všeobecnosti sa dá povedať, že PND reprezentuje blízkosť prekážok vzhľadom na ťažisko robota a RND reprezentuje blízkosť prekážok od hranice robota.



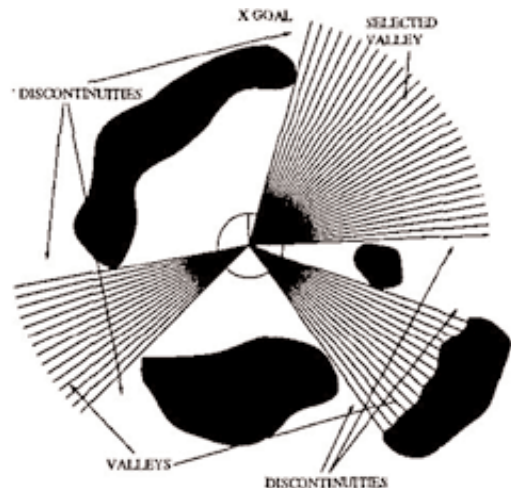
Obr.6 Definícia hraníc, volných regiónov a vybraného regiónu [4]

Po vytvorení diagramov sa vykonáva ich analýza. Pri PND diagrame sa najskôr nájdu hranice v prostredí (medzi voľným a obsadeným priestorom). Z týchto hraníc možno definovať voľné úseky a podľa istého kritéria jeden z nich vybrať. V PND diagrame možno identifikovať body nespojitosti. Medzi dvoma susednými sektormi vzniká bod nespojitosti, ak je výškový rozdiel v PND väčší ako  $l$ . Ďalej možno v PND identifikovať údolia. Množina sektorov  $S = \{s_i\}_{i=1, \dots, k}$  vytvára údolie, ak platia

tieto dve podmienky:

1. Všetky sektory z  $S$  sú susedné a nie je medzi nimi bod nespojitosti.
2. Nech  $s_l$  a  $s_r$  sú krajné sektory z množiny  $S$ . Nech  $s_{nl}$  a  $s_{nr}$  sú susedné sektory sektorov  $s_l$  a  $s_r$  nepatriace do množiny  $S$ . Potom platí:

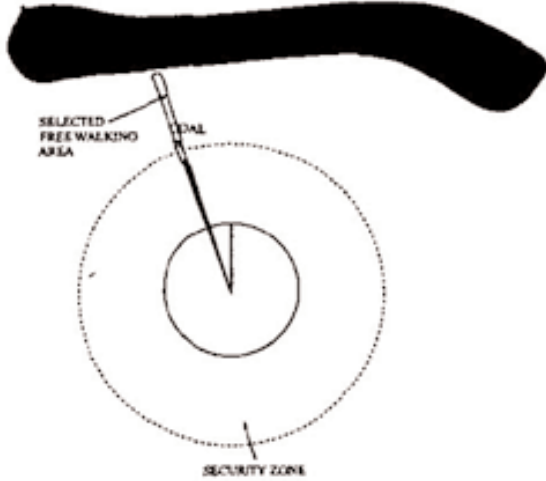
$$PND_{s_{nl}} - PND_{s_l} > l \wedge PND_{s_{nr}} - PND_{s_r} > l$$



Obr.7 Definícia bodov nespojitosti, údolí a vybraného údolia [4]

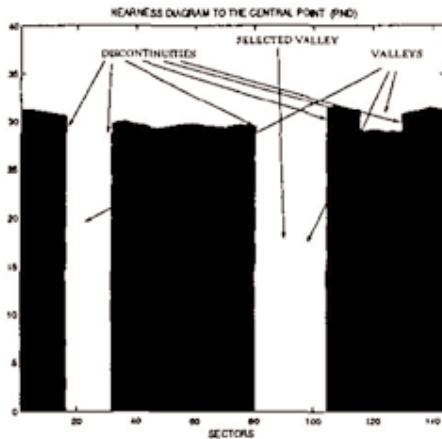


Špeciálny prípad nastáva, ak sa cieľ nachádza medzi robotom a prekážkou, teda je ešte pred prekážkou. Vtedy je sektor v smere cieľa nastavený na nulu a vznikne tak umelé údolie (obr. 8).



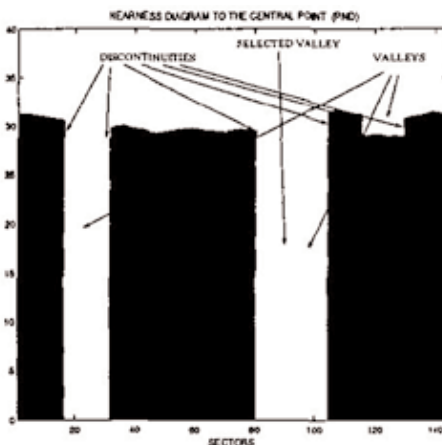
Obr.8 Prípad vytvoreného umelého údolia [4]

Treba si všimnúť, že pri definovaní údolí možno definovať aj rastúce a klesajúce body nespojitosti. Údolia možno rozdeliť medzi široké a úzke na základe rozdeľovacieho parametra  $S_{max}$ . Výber údolia spočíva v selekcii takého údolia, ktoré má najmenšiu sektorovú vzdialenosť od sektora v smere cieľa.



Obr.9 PND Nearness diagram [4]

Analýza diagramu RND spočíva predovšetkým v zadenovaní pojmov bezpečná vzdialenosť a bezpečná blízkosť. Bezpečná vzdialenosť je minimálna tolerovateľná vzdialenosť  $d_s$  od prekážky (od nej nie je zaručená bezpečnosť robota). Bezpečná blízkosť je maximálna tolerovateľná blízkosť  $n_s$  od prekážky. Je to hodnota z RND, keď je robot príliš blízko k prekážke a počíta sa ako  $n_s = d_{max} - d_s$ .



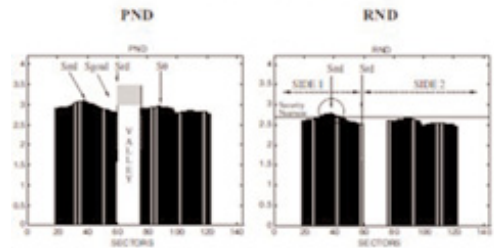
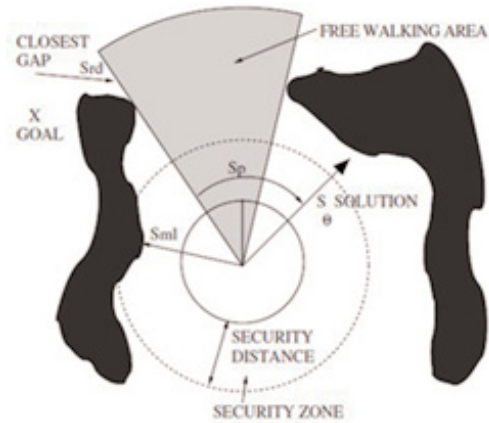
Obr.10 RND Nearness diagram [4]

Robot sa potom môže nachádzať v stave s vysokou bezpečnosťou (HS) alebo v stave s nízkou bezpečnosťou (LS). Stav LS je dosiahnutý, ak aspoň jeden sektor v RND prekročí bezpečnú blízkosť. To znamená, že vzdialenosť medzi hranicou robota a prekážkou je menšia ako bezpečná vzdialenosť. Ak sa robot nenachádza v stave LS, potom je v stave HS.

Ak sú oba diagramy takto analyzované, možno zadenovať päť všeobecných situácií, ktoré môžu nastať:

1. LS1 (Low Safety 1)

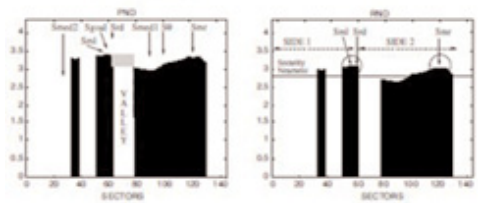
Robot je v tomto stave, ak aspoň jeden sektor prekročil bezpečnú blízkosť v RND na strane rastúcej nespojitosti v smere definovanom od tejto nespojitosti k cieľu.



Obr.11 Príklad LS1 [5]

2. LS2 (Low Safety 2)

Robot je v tomto stave, ak aspoň jeden sektor prekročil bezpečnú blízkosť v RND na oboch stranách rastúcej nespojitosti.

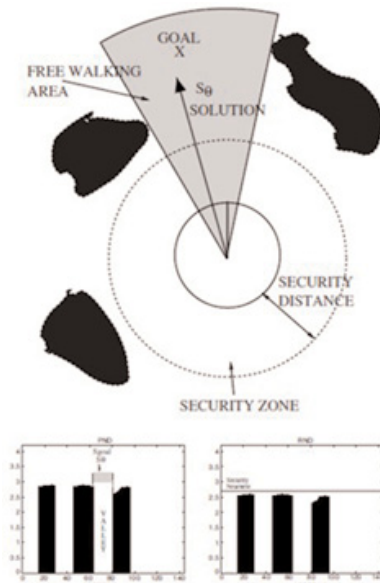


Obr.12 Príklad LS2 [5]



3. HSGV (High Safety Goal in Valley)

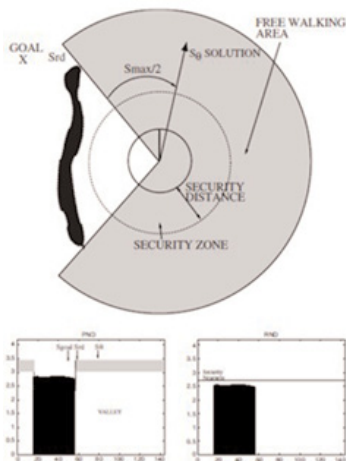
Robot je v tomto stave, ak vybrané údolie obsahuje cieľový sektor.



Obr.13 Príklad HSGV [5]

4. HSWV (High Safety Wide Valley)

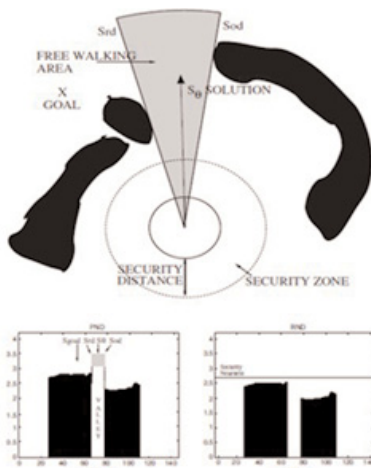
Robot je v tomto stave, ak je vybrané údolie definované ako široké a cieľ sa nenachádza v tomto údolí.



Obr.14 Príklad HSWV [5]

5. HSNV (High Safety Narrow Valley)

Robot je v tomto stave, ak je vybrané údolie definované ako úzke a cieľ sa nenachádza v tomto údolí.



Obr.15 Príklad HSNV [5]

Stratégie navigácie pri tejto metóde sú odvodené práve od týchto všeobecných situácií. Keďže je Nearness diagram rýchlostnou metódou, výstupom je opäť usporiadaná dvojica  $(\omega, v)$ . Cieľom je však najskôr vypočítať zvolený sektor a na základe tohto sektora zvoliť požadovanú rýchlosť. Volba sektora riešenia  $s_\theta$  závisí od typu všeobecnej situácie:

1. LS1

Pri LS1 presahujú prekážky do bezpečnostnej zóny len na jednej strane od vybraného údolia, tej bližšej k cieľu. Riešenie teda musí robotu predpísať taký pohyb, ktorým sa prekážky dostanú mimo dosahu bezpečnostnej zóny. Preto je sektor riešenia určený:

$$s_p = |s_{rd} - s_{ml}|p + \frac{s_{max}}{2}$$

$$s_\theta = s_{rd} + sign(s_{rd} - s_{ml})s_p \tag{16}$$

kde  $s_{rd}$  je sektor zodpovedajúci bodu rastúcej nespojitosti (z angl. rising discontinuity) najbližšieho k sektoru cieľa  $s_{goal}$  vybraného údolia,  $s_{ml}$  je sektor s najvyššou hodnotou v RND, ktorý prekračuje bezpečnú blízkosť na strane rastúcej nespojitosti vybraného údolia (zodpovedá najbližšej prekážke) a  $p$  je experimentálne nastavený parameter, ktorého veľkosť ovplyvňuje prechod medzi situáciami a plynulosť pohybu robota (obvykle volený medzi 1.5 a 2.5). Cieľom tejto stratégie je teda dostať robot čo najďalej od blízkej prekážky a zároveň sa dostať k medzere bližšej k cieľu.

2.LS2

V tejto situácii prekážky presahujú do bezpečnostnej zóny na oboch stranách vybraného údolia. Preto musí byť robotu predpísaný pohyb, ktorý vycentruje robot medzi tieto prekážky, zatiaľ čo sa musí pohnúť k medzere bližšej k cieľu. Sektor riešenia je potom určený:

$$s_{med1} = \frac{s_{ml} + s_{mr}}{2}$$

$$s_{med2} = \frac{s_{ml} + s_{mr} + n}{2}$$

$$s_\theta = s_{med1} \pm c \begin{cases} |s_{rd} - s_{med1}| < |s_{rd} - s_{med2}| \\ \text{inak} \end{cases}$$

$$s_\theta = s_{med2} \pm c \tag{17}$$

kde  $s_{ml}$  a  $s_{mr}$  sú sektory s najvyššou hodnotou z RND, ktoré prekračujú bezpečnú blízkosť na oboch stranách vybraného údolia (zodpovedajú dvom najbližším prekážkam),  $c$  je korekcia využitá na vycentrovanie robota medzi prekážky a  $n$  je počet sektorov. Korekcia  $c$  závisí od vzdialenosti bližšej prekážky a od rozdielu vzdialeností zodpovedajúcich dvoch blízkych prekážok. Táto hodnota je pripočítaná alebo odpočítaná vo funkcii výberu sektora tak, aby robot dosiahol rovnakú vzdialenosť od oboch najbližších prekážok.

3.HSGV

Keďže sa v tomto prípade cieľový sektor nachádza vo vybranom údolí, sektor riešenia je určený:

$$s_\theta = s_{goal} \tag{18}$$

4.HSWV

Pri tejto situácii sa cieľový sektor nenachádza vo vybranom údolí, preto je riešením sledovanie obrysu prekážky. Sektor riešenia je teda určený:

$$s_\theta = s_{rd} \pm \frac{s_{max}}{2} \tag{19}$$

kde  $s_{rd}$  je sektor zodpovedajúci bodu rastúcej nespojitosti vybraného údolia bližšie k cieľu.



## 5.HSNV

Podobne ako v predchádzajúcom prípade cieľ sa nenachádza vo vybranom údolí. Rozdiel je však v tom, že v tomto prípade je toto údolie úzke. Sektor riešenia sa potom vypočíta ako:

$$s_{\theta} = \frac{s_{rd} + s_{od}}{2} \quad (20)$$

kde  $s_{rd}$  a  $s_{od}$  sú sektory nespojitostí z okrajov vybraného údolia.

Samotná predpísaná translačná rýchlosť je potom predpísaná na základe typu všeobecnej situácie. Nech je  $v_{max}$  maximálna translačná rýchlosť robota,  $d_{obs}$  vzdialenosť k najbližšej prekážke od hranice robota,  $d_s$

bezpečná vzdialenosť a  $\theta$  uhol odvodený od  $s_{\theta}$  z intervalu  $\left\langle -\frac{\pi}{2}, \frac{\pi}{2} \right\rangle$ .

Pre všeobecné situácie s vysokou bezpečnosťou platí:

$$v = v_{max} \left| 1 - \frac{\theta}{\frac{\pi}{2}} \right| \quad (21)$$

Pre všeobecné situácie s nízkou bezpečnosťou platí:

$$v = v_{max} \frac{d_{obs}}{d_s} \left| 1 - \frac{\theta}{\frac{\pi}{2}} \right| \quad (22)$$

Predpisom takéhoto správania sa robot pohybuje v prostredí s maximálnou rýchlosťou, kým nejaká prekážka neprekročí bezpečnostnú zónu. Potom robot zníži svoju rýchlosť v závislosti od blízkosti tejto prekážky a snaží sa dostať mimo dosahu tejto prekážky v bezpečnostnej zóne. Navyše priebeh pohybu je plynulý.

Nech  $\omega_{max}$  je maximálna uhlová rýchlosť, potom predpísaná uhlová rýchlosť robota:

$$\omega = \omega_{max} \frac{\theta}{\frac{\pi}{2}} \quad (23)$$

Ak je odchýlka medzi skutočnou a predpísanou hodnotou natočenia robota veľká, tento vzťah predpíše robotu pohyb s veľkým zatočením. Naopak ak je odchýlka malá, robot vykoná plynulý pohyb.

### Podakovanie

Tento článok vznikol pri riešení projektu VEGA 1/0690/09 a KEGA 3/7307/09.

### Literatúra

- [1] SIEGWART, R. – NOURBAKHSI I.R.: Introduction to Autonomous Mobile Robots. Massachusetts Institute of Technology, 2004. ISBN-13 978-0-262-19502-7.
- [2] FOX, D. – BURGARD, W. – THRUN, S.: The Dynamic Window Approach to Collision Avoidance. IEEE Robotics & Automation Magazine, 1070-9932/97/\$10.0001997 IEEE, p.23-33.
- [3] SIMMONS, R.: The Curvature-Velocity Method for Local Obstacle Avoidance. Proceedings of the 1996 IEEE International Conference on Robotics and Automation, Minneapolis, Minnesota, April 1996.
- [4] MINGUEZ, J. – MONTANO, L.: Nearness Diagram Navigation (ND) (A New Real Time Collision Avoidance Approach). Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems, p. 2094 – 2100.

[5] MINGUEZ, J. – MONTANO, L.: Nearness Diagram (ND) Navigation (Collision Avoidance in Troublesome Scenarios). IEEE Transactions on Robotics and Automation, vol. 20, no. 1, February 2004, p. 45 – 59.

[6] VIKENMARK, D.: The Obstacle-Restriction Method (ORM) for Reactive Obstacle Avoidance in Difficult Scenarios in Three-Dimensional Workspaces. Master of Science Thesis, Stockholm, Sweden 2006. ISSN-1653-5715.

prof. Ing. Ladislav Jurišica, PhD.  
Ing. František Duchoň, PhD.  
Ing. Andrej Babinec

Fakulta elektrotechniky a informatiky  
Ústav riadenia a priemyselnej informatiky  
Ilkovičova 3, 812 19 Bratislava  
e-mail: ladislav.juristica@stuba.sk  
frantisek.duchon@stuba.sk  
andrej.babinec@stuba.sk