

# Modelovanie a riadenie mobilného robota

František Duchoň, Roman Murár

## Abstrakt

Tento článok je o modelovaní, riadení a navigácii mobilného robota v neznámom štruktúrovanom prostredí. Výsledkom sú experimentálne simulácie, ktoré slúžili na porovnanie dvoch metód lokálneho plánovania dráhy mobilného robota.

**Kľúčové slová:** modelovanie mobilného robota, riadenie mobilného robota, navigácia mobilného robota, mapa prostredia, potenciálove polia, metóda napnutého vlákna

## Úvod

Pre zabezpečenie zmysluplnej činnosti robota vo vnútorných prostrediach potrebujeme poznať adekvátnu reprezentáciu jeho pracovného prostredia, tzv. mapu prostredia. Základnými požiadavkami na túto reprezentáciu sú abstraktnosť a dostatočná detailnosť. Okrem týchto požiadaviek je požiadavkou na pohyb robota aj bezpečnosť (robota samotného, ale aj prostredia), ktorá je závislá na schopnosti robota vnímať svoje okolie svojimi senzormi.

## 1. Tvorba mapy prostredia

Robot pozná svoju štartovaciu polohu a jej orientáciu, cieľovú polohu a jej orientáciu, na základe čoho vieme generovať jeho pohyb. Pri orientácii v prostredí sa bežne používajú ultrazvukové alebo infračervené diaľkomery, ktoré vysielajú signál [9]. Z takýchto snímačov robot prijíma údaje, ktoré sa týkajú vzdialeností k najbližším prekážkam.

Väčšina robotov je schopných vypočítať prejdenú vzdialenosť a orientáciu zo snímačov pohybového systému – odometria [1] [2]. Pomocou odometrie môže robot aktualizovať svoju polohu a orientáciu vo vytváraní mapy prostredia, a preto sa môžu nové merania snímačov zaznamenať do tejto mapy relatívne k novej pozícii. Problémom sú však vnášané chyby (z odometrie, ale aj z merania senzorov). Napríklad pri použití ultrazvukového snímača poznáme len hodnotu vzdialenosti k najbližšej prekážke v laloku daného snímača, čiže nepoznáme ani jej smer ani jej presnú polohu.

Mapa prostredia teda nie je presná, čiže je neurčitá. Preto sa používa mapovanie, čo je činnosť založená na filtrovaní mapy prostredia. Jednou z metód filtrovania prostredia je aj metóda reprezentácie prostredia pomocou mriežok obsadenia. Mriežky obsadenia sa používajú pre dvojrozmerné vyjadrenie usporiadania prostredia. Každá bunka mriežky reprezentuje určitú oblasť prostredia a obsahuje aj informáciu o subpriestore, ktorý reprezentuje (najčastejšie: voľný priestor, prekážka). Najčastejšie sa používajú metódy využívajúce pravdepodobnostné modely snímačov. Jednou z nich je metóda navrhnutá Sebastianom Thrunom - učiace sa mapy pre navigáciu mobilného robota v uzavretých priestoroch.

### 1.1 Učiace sa mapy pre navigáciu mobilného robota v uzavretých priestoroch

Táto metóda používa aktualizáciu mapy odvodenú od Bayesovho vzťahu:

$$P(A|B) = P(B|A) \frac{P(A)}{P(B)}, \quad (1)$$

a na vytvorenie mriežkovej mapy používa integráciu mnohých snímaní senzormi využívajúc neurónové siete [11]. Neurónová sieť bola natrénovaná (spätnoväzobná) na interpretovanie snímania a zapísania hodnôt obsadenia do buniek v mape (výstup: 1 – obsadené, 0 – voľná, 0,5 - neistota). Tento prístup sa skombinuje s predchádzajúcou znalosťou o obsadení buniek pomocou Bayesovho pravidla na získanie novej hodnoty pravdepodobnosti obsadenia jednotlivých buniek. Použitie neurónových sietí namiesto výpočtových metód má dve hlavné výhody. Prvou je, že ak sa prostredie drasticky zmení, sieť môže byť rýchlo natrénovaná. Druhou výhodou je, že mnohonásobné snímanie je vyhodnocované v rovnakom čase oproti uvažovaniu každého snímača za nezávislého od iných, čo môže byť užitočné pri identifikovaní chybných snímaní.

Pre hociaké snímanie potrebujeme najst'  $\text{Prob}(\text{occ}_{x,y}|s^{(t)})$ , čo je pravdepodobnosť, že bunka  $(x,y)$  je obsadená a táto informácia bola určená snímaním  $s$  v čase  $t$ . Pre interpretáciu tohto snímania Thrun natrénoval neurónovú sieť so spätnou väzbou pre jedno snímanie ultrazvukovým snímačom v rozsahu hodnôt  $[0,1]$ . V ďalšom kroku treba integrovať túto hodnotu s predchádzajúcou:

$$\text{Prob}(\text{occ}_{x,y}|s^{(1)}, \dots, s^{(T)}) = 1 - \left( 1 - \frac{\text{Prob}(\text{occ}_{x,y}) \prod_{t=1}^T \text{Prob}(\text{occ}_{x,y}|s^{(t)}) (1 - \text{Prob}(\text{occ}_{x,y}))}{1 - \text{Prob}(\text{occ}_{x,y}) \prod_{t=1}^T (1 - \text{Prob}(\text{occ}_{x,y}|s^{(t)})) \text{Prob}(\text{occ}_{x,y})} \right)^{-1}, \quad (2)$$

kde  $\text{Prob}(\text{occ}_{x,y})$  je predchádzajúca hodnota pravdepodobnosti obsadenia bunky, ktorá môže byť v prvom kroku prednastavená na hodnotu 0,5. Tento vzťah môže byť odvodený z Bayesovho vzťahu využívajúc predpoklad podmienenej pravdepodobnosti, ktorý hovorí, že  $\text{Prob}(s^{(t)}|\text{occ}_{x,y})$  je nezávislá od  $\text{Prob}(s^{(t-1)}|\text{occ}_{x,y})$  z čoho vyplýva, že pravdepodobnosť získania snímania v určitom čase, ktoré hovorí o pravdepodobnosti obsadenia buniek v mape, je nezávislá od predchádzajúcich snímaní.

Bayesov vzťah hovorí:

$$\frac{\text{Prob}(\text{occ}_{x,y}|s^{(1)}, \dots, s^{(T)})}{\text{Prob}(\text{occ}_{x,y}|s^{(1)}, \dots, s^{(T-1)})} = \frac{\text{Prob}(s^{(T)}|\text{occ}_{x,y}, s^{(1)}, \dots, s^{(T-1)}) \cdot \text{Prob}(\text{occ}_{x,y}|s^{(1)}, \dots, s^{(T-1)})}{\text{Prob}(s^{(T)}|\text{occ}_{x,y}, s^{(1)}, \dots, s^{(T-1)}) \cdot \text{Prob}(\text{occ}_{x,y}|s^{(1)}, \dots, s^{(T-1)})}, \quad (3)$$

čo môže byť zjednodušené využitím predpokladu podmienenej nezávislosti:

$$= \frac{\text{Prob}(s^{(T)}|\text{occ}_{x,y}) \cdot \text{Prob}(\text{occ}_{x,y}|s^{(1)}, \dots, s^{(T-1)})}{\text{Prob}(s^{(T)}|\text{occ}_{x,y}) \cdot \text{Prob}(\text{occ}_{x,y}|s^{(1)}, \dots, s^{(T-1)})}. \quad (4)$$

Aplikovaním Bayesovho vzťahu na vzťah (3):

$$= \frac{\text{Prob}(\text{occ}_{x,y} | s^{(t)}) \cdot \text{Prob}(\overline{\text{occ}}_{x,y}) \cdot \text{Prob}(\text{occ}_{x,y} | s^{(1)}, \dots, s^{(T-1)})}{\text{Prob}(\overline{\text{occ}}_{x,y} | s^{(t)}) \cdot \text{Prob}(\text{occ}_{x,y}) \cdot \text{Prob}(\overline{\text{occ}}_{x,y} | s^{(1)}, \dots, s^{(T-1)})} \quad (5)$$

Indukciou nad T dostaneme:

$$= \frac{\text{Prob}(\text{occ}_{x,y}) \cdot \prod_{t=1}^T \frac{\text{Prob}(\text{occ}_{x,y} | s^{(t)}) \cdot (1 - \text{Prob}(\text{occ}_{x,y}))}{1 - \text{Prob}(\text{occ}_{x,y}) + \text{Prob}(\text{occ}_{x,y} | s^{(t)}) \cdot \text{Prob}(\text{occ}_{x,y})}}{1 - \text{Prob}(\text{occ}_{x,y}) + \text{Prob}(\text{occ}_{x,y} | s^{(1)}, \dots, s^{(T)})} \quad (6)$$

Vzťah (2) dostaneme využitím skutočnosti, že:

$$\text{Prob}(\overline{\text{occ}}_{x,y} | s^{(1)}, \dots, s^{(T)}) = 1 - \text{Prob}(\text{occ}_{x,y} | s^{(1)}, \dots, s^{(T)}) \quad (7)$$

Ak snímanie potvrdí, že bunka je obsadená a predchádzajúca hodnota pravdepodobnosti obsadenia tiež hovorila o tom, že je obsadená, tak nová pravdepodobnosť obsadenia je vyššia. Ak si predchádzajúca hodnota a nasnímaná hodnota odporujú, hodnota sa blíži k 0,5. Ak sa opäť potvrdí, že je voľná, tak hodnota sa blíži k hodnote 0. V tejto metóde sa používajú dve neuronové siete, ktoré zovšeobecňujú skúsenosti z reálneho sveta. Jedna interpretuje snímanie senzorov a druhá vyhodnocuje spokojnosť s týmto snímaním. Obidve siete vlastne kódujú charakteristiku snímačov a typické prostredie, s ktorým sa robot môže stretnúť, napríklad rozmanité kancelárske prostredie.

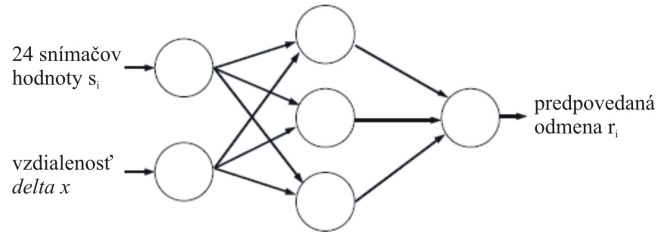
V každom časovom kroku  $i$  obe tieto siete vezmú snímania senzorov ako vstupy, nazvané vnemy, a polohu robota  $x_i$ .

Výsledkom je očakávaná odmena  $r_i$  (negatívna alebo pozitívna), získaná pri pohybe do pozície  $x$ . Adaptívny model,  $M$ , zovšeobecňuje množinu príkladov  $\{(x_i, s_i, r_i) | i = 1..n\}$  na nové pozície v oblasti  $r = M(\{(x_i, s_i, r_i) | i = 1..n\}, x)$ , kde  $r$  je odmena (pokuta) pri pohybe do oblasti  $x$ . Ak je teda získané nové snímanie, sieť na interpretovanie snímača, R (z angl. reading, Obr. 1), vytvorí novú lokálnu mapu obklopujúcu robota. Sieť vyhodnocujúca spokojnosť so snímaním, C (z angl. confidence, Obr. 2), predpovedá platnosť každého snímania a oslabuje vplyv snímaní s nízkou hodnotou spokojnosti v lokálnej mape. Takto vytvorená lokálna mapa je potom integrovaná do globálnej mapy.

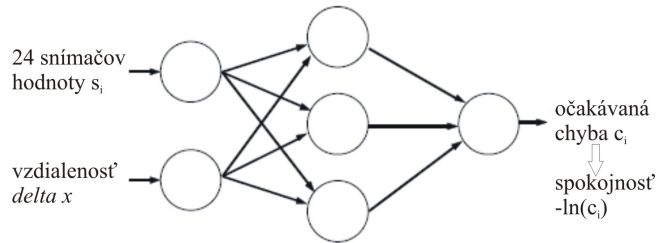
Obidve siete R (sieť na interpretovanie snímania) a C (sieť na odhad spokojnosti so snímaním) boli natrénované spätným šírením a supervízorovým učením, využívajú jednu skrytú vrstvu obsahujúcu osem neurónov. Siete boli natrénované v známom prostredí, kde každá prekážka dávala negatívnu odmenu, ak robot do nej mohol naraziť. Nevýhodou sú obmedzenia sietí, ktoré nie vždy dobre zovšeobecňujú, čo môže v neznámych prostrediach, na ktoré neboli natrénované viesť ku kolíznym situáciám. Avšak podľa autora tejto metódy, tréning bolo efektívne a siete úspešne zovšeobecňovali v skutočných kancelárskych prostrediach.

Sieť na interpretáciu snímania zakresľuje jednu pozíciu a sadu snímaní s dôsledkom odmenenia danej bunky  $x_i$ . Vstupmi sú teda snímania senzorov a vzdialenosť  $x_i$  od súčasnej pozície robota,  $x$ . Výstupom je potom očakávaná odmena (alebo pravdepodobnosť obsadenia) danej buny  $x_i$ .

Mohlo by sa zdať, že táto funkcia odmeňovania pomocou neuronovej siete vracia jednu odmenu,  $r$ , pre každú pozíciu a sadu snímaní. Presnejšie však vracia  $n$  odhadov odmien, jednu pre každý údajový bod  $(s_i, x_i)$ . Toto v podstate znamená, že celá mapa je aktualizovaná v každom časovom kroku  $t_i$ . Potom teda aj snímače, ktoré nič nenasnímali budú propagované do nekonečna a body za múrmi budú aktualizované ako prázdne. Preto boli vymyslené techniky, ktoré odstránili tieto dve zásadné chyby.



Obr.1 Sieť na interpretáciu snímania, R  
Fig.1 Network for scanning interpretation, R



Obr.2 Sieť na odhad spokojnosti so snímaním, C  
Fig.2 Network for satisfaction estimation, C

Jednou z nich je sieť na odhad spokojnosti so snímaním, C. Je to teda druhá sieť, ktorá pracuje paralelne so sieťou R. Vstupom do tejto siete sú rovnaké vstupy ako do siete R.

Použitie  $\Delta x$  ako vstupu odstráni potrebu modelovania globálneho súradnicového systému pre sieť, čo významne zníži výpočtový čas. C bola natrénovaná na meranie očakávanej chyby funkcie odmeňovania R. Akonáhle bola natrénovaná sieť R, na natrénovanie siete C bol použitý nezávislý test na odhad očakávanej chyby medzi funkciou odmeňovania R bunky  $x$  a jej skutočnou hodnotou obsadenia. Ak bola chyba veľká, naša spokojnosť s hodnotou vrátenou sieťou R pre bunku  $x$  je nízka, a naopak. Snímania z R,  $r_i$ , sú potom váhované s príslušnými mierami spokojnosti, definovanými ako  $-\ln(c_i)$  a združenými pomocou vzťahu:

$$M(x) = \frac{\sum_i -\ln c_i(x) \cdot r_i(x)}{c_M(x)} \quad (8)$$

kde  $c_M(x) = \sum_i -\ln c_i(x)$  je nazývaná súhrnná spokojnosť

v bunke  $x$ . Tento vzťah teda slúži ako normalizácia. Podľa tohto vzťahu by sa teda mala aktualizovať celá mapa v každom časovom kroku, čo by viedlo k veľkej výpočtovej náročnosti. Namiesto toho sa teda využíva maximálny rozsah snímania senzormi a aktualizujú sa len body v tomto rozsahu, čo vedie k značnému urýchleniu výpočtov.

Nevýhodou samotnej siete C je, že si nepamätá žiadne predchádzajúce hodnoty spokojnosti so snímaním. Napríklad ak snímač 1 zosnímal veľkú vzdialenosť a dostal nízky odhad spokojnosti kvôli snímaču 2, ktorý zosnímal malú vzdialenosť, je celkom jasné, že snímanie senzora 1 bolo zrkadlovým odrazom.

Avšak ak v ďalšom časovom kroku, snímač 2 zosnímal veľkú vzdialenosť kvôli diere v stene alebo nejakému šumu, snímač 1 bude ohodnotený vysokou hodnotou spokojnosti. Táto nedokonalosť tejto metódy sa dá odstrániť metódou budúcej predpovedi, ktorá odhaduje pozíciu a orientáciu objektov v mape, ktorá časom slabne, avšak „neodhadzuje“ ju ako sieť na odhad spokojnosti zo snímaním.

## 2. Plánovanie pohybu mobilného robota

Plánovanie bezpečného (bezkolízneho) pohybu robota v prostredí s prekážkami je jednou zo základných úloh, ktoré treba riešiť, aby robot mohol vykonávať určené činnosti. Cieľom je navigovať robot zo štartu do cieľa v prostredí s prekážkami tak, aby generovaná dráha nekolidovala so žiadnou prekážkou [8].

V reálnom svete však nemáme detailné informácie o prostredí, a to sa navyše môže v čase meniť. Prostredie môže byť teda dynamické alebo neznáme, preto sa používa lokálne plánovanie (resp. reaktívna navigácia), ktorého vstupom sú informácie zo senzorov.

Pre riešenie pohybu autonómneho robota existuje množstvo metód. Medzi metódy globálneho plánovania trajektórie patria potenciálové metódy (potenciálové polia, metóda voľného koridoru, metóda sledovania steny prekážky), cestná mapa, dekompozícia na bunky, dištančná transformácia a hľadanie grafov [5] [3] [1] [2]. Medzi metódy lokálneho plánovania trajektórie patrí konfigurácia priestoru, priestor pokrytý mriežkou a potenciálové polia.

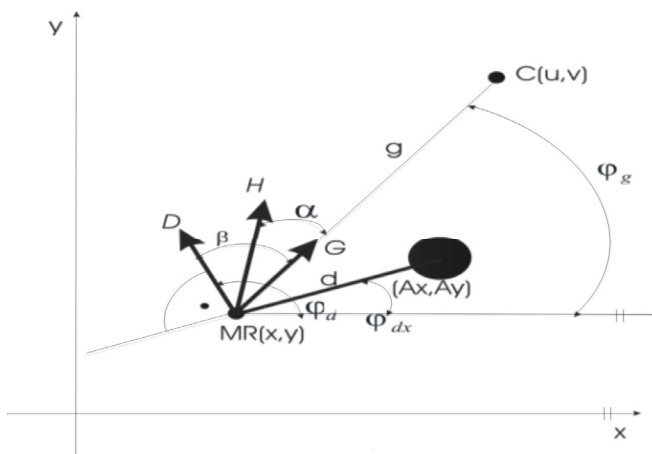
Metódy lokálneho plánovania (reaktívne metódy) možno rozdeliť do dvoch skupín. Prvá skupina je založená na modeli (napr. metóda umelého potenciálového pola) a využívajú istý spôsob reprezentácie aktuálneho modelu prostredia. Druhú skupinu tvoria metódy založené na prostredí používajú aktuálne senzorové informácie [8].

V prípade uplatnenia lokálnych metód plánovania, sa môže robot dostať do situácie tzv. lokálneho minima (dochádza k zastaveniu robota na mieste). Na odstránenie tohto problému sa používajú metódy, ktoré si pamätajú aj predchádzajúce stavy, prepnutie na iné správanie, sledovanie steny alebo využitie hornej úrovne riadenia [10].

## 3. Potenciálové polia

Metóda plánovania dráhy metódou potenciálových polí spočíva v zadefinovaní dvoch síl [3] [8] (Obr. 3):

1. repulzívna sila (D) – reprezentácia sily, ktorá mobilný robot odpudzuje od prekážok,
2. atraktívna sila (G) – reprezentácia sily, ktorá mobilný robot priťahuje do cieľa.



Obr.3 Metóda potenciálových polí

Fig.3 Potential field method

Princíp metódy je založený na výpočte výslednej sily (H). Na výpočet výslednej sily je potrebné vedieť vzdialenosť od cieľa (g) a minimálnu vzdialenosť od prekážky (d). Vlastný

výpočet vidieť na vzťahoch (9) až (14). Nevýhodou tejto metódy je existencia lokálnych minim, čo môže viesť k nesprávnemu cieľu. Štandardne pri lokálnych metódach meriame uhly proti smeru hodinových ručičiek a prekážky sa snažíme obchádzať podľa pravidiel cestnej premávky. Je to však len vec dohody.

$$\varphi_d = 90 + \varphi_{dx} = 90 + \arctg \left( \frac{A_y - y}{A_x - x} \right), \quad (9)$$

$$\varphi_g = \arctg \left( \frac{v - y}{u - x} \right), \quad (10)$$

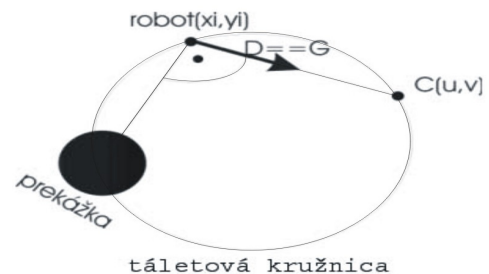
$$\beta = \varphi_d - \varphi_g, \quad (11)$$

$$H : \alpha = \beta * f = \beta * \left( 1 - \frac{d}{g} \right), \quad (12)$$

$$y_{i+1} = y_i + k(d_i) * \sin[\varphi_g - (\varphi_d - \varphi_g) * f_i], \quad (13)$$

$$x_{i+1} = x_i + k(d_i) * \cos[\varphi_g - (\varphi_d - \varphi_g) * f_i], \quad (14)$$

kde  $k(d_i)$  je funkcia závislá od vzdialenosti k prekážke. Otázkou, kedy sa už robot môže pohybovať po priamej trajektórii do cieľa rieši zostrojenie táletovej kružnice podľa Obr. 4.



Obr.4 Priame smerovanie do cieľa

Fig.4 Direct movement to target

Pre zlepšenie plánovania trajektórie pomocou metódy potenciálových polí sa používajú rôzne modifikácie tejto metódy [3]. Vo väčšine prípadov ide o úpravu deliaceho pomeru f. Príklady takýchto vylepšení:

$$\text{vylepšenie: } f = 1 - \frac{d}{g} \frac{g_0}{d_0}, \quad (15)$$

kde  $g_0, d_0$  sú konštanty (na začiatku).

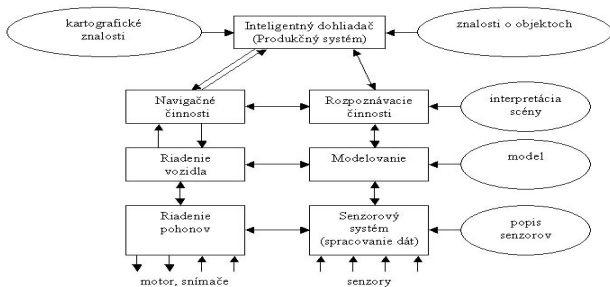
$$\text{komplik. vylepšenie: } f = 1 - \frac{d}{g} - \frac{g_0 - A * x}{k} \frac{d}{g}, \quad (16)$$

kde A určuje ako ďaleko "vyletiť" od prekážky, x je aktuálna pozícia a k je heuristická konštanta.

## 4. Modelovanie činnosti mobilných robotov

Modelovanie činnosti mobilných robotov môžeme rozdeliť do štyroch úrovní [3] (Obr. 5). Prvá (najnižšia) úroveň spočíva v návrhu senzorickeho a pohonného systém, a najmä v ich modelovaní. Druhá úroveň sa zaoberá tvorbou modelu činnosti robota. Tretia úroveň ma za úlohy rozpoznávacie a navigačné činnosti. Najvyššou úrovňou je inteligentný dohliadač, ktorý využíva kartografické znalosti a znalosti o objektoch.

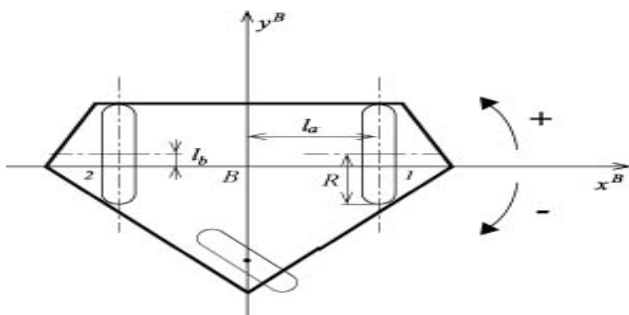
Vyššie úrovne riadenia zabezpečujú bezkolízne správanie sa mobilného robotického systému. V prípade, keď nastane situácia „lokálneho“ minima zabezpečuje vymanévrovanie z tejto situácie už vyššie opísanými metódami (pamäť predchádzajúcich stavov, prepnutie na iné správanie, sledovanie steny, generovanie čiastkových cieľov) [10].



**Obr.5 Modelovanie činnosti mobilných robotov**  
**Fig.5 odelling of mobile robot functions**

## 5. Modelovanie robota

Modelovaný je robot (Obr. 6) s trojkolesovým podvozkom s riadenými zadnými kolesami a otočným neriadeným predným kolesom (dekomponovaný na dvojkolesový podvozok) [4]. Robot má dva stupne voľnosti - pohyb v smere osi y a alebo pohyb v smere osi x alebo rotačný pohyb.



**Obr.6 Modelovaná kinematická schéma robota**  
**Fig.6 Modelling of mobile robot kinematic scheme**

Pre tento prípad možno odvodiť rovnicu :

$$\begin{pmatrix} v_{Bx} \\ v_{By} \\ \omega_{Bc} \end{pmatrix} = \frac{R}{2l_a} \begin{pmatrix} -l_b & l_b \\ -l_a & -l_a \\ -1 & 1 \end{pmatrix} \begin{pmatrix} \omega_{W1} \\ \omega_{W2} \end{pmatrix}, \quad (17)$$

a inverznú rovnicu:

$$\begin{pmatrix} \omega_{W1} \\ \omega_{W2} \end{pmatrix} = \frac{1}{R(l_b^2 + 1)} \begin{pmatrix} -l_a l_b & -l_b^2 - 1 & -l_a \\ l_a l_b & -l_b^2 - 1 & l_a \end{pmatrix} \begin{pmatrix} v_{Bx} \\ v_{By} \\ \omega_{Bc} \end{pmatrix}, \quad (18)$$

kde  $v_{Bx}, v_{By}$  sú translačné rýchlosti robota v smere prísluš-

ných osí [m.s<sup>-1</sup>],  $\omega_{Bc}$  je uhlová rýchlosť rotácie robota [rad.s<sup>-1</sup>],  $\omega_{W1}, \omega_{W2}$  sú uhlové rýchlosti kolies robota [rad.s<sup>-1</sup>],

$R$  je polomer hnacích kolies [m],  $l_a, l_b$  sú vzdialenosti kolies od osí robota [m]. V prípade modelovania snímačov a samotného rozpoznávania prostredia sme si situáciu zjednodušili na model reprezentujúci stopercentnú znalosť o polohe a orientácii jednotlivých prekážok v lokálnej mape prostredia.

Vlastný algoritmus spočíva z vyhodnocovania situácie robota v neznámom štruktúrovanom prostredí. Najskôr sa vyhodnotí, či má v smere cieľa nejakú stacionárnu prekážku. Ak nie, nasmeruje sa priamo do cieľa. Podľa vzťahu

$$\varphi = \arctg\left(\frac{A_y - y}{A_x - x}\right) \quad (\text{prípadne podľa kvadranta k cieľu}) \text{ sa určí}$$

želané natočenie robota, kde  $A_y, A_x$  sú súradnice cieľa a  $x, y$  je pozícia robota.

Ak sa robot nachádza v blízkosti prekážky, ale má voľnú cestu do cieľa, tak len spomalí, v prípade nutnosti vymanérovania od prekážky. Ak sa dostane do tesnej blízkosti pre-

kážky, ktorá je v smere cieľa, nastupuje úroveň algoritmu obchádzania prekážky. Spočíva vo využití logických flagov a vo využití senzorického systému. Robot sa snaží obísť prekážku modifikovanou metódou napnutého vlákna, čiže natočenie robota sa mení tak, aby robot šiel vždy na prekážku kolmo (minimum je na krajných senzoch).

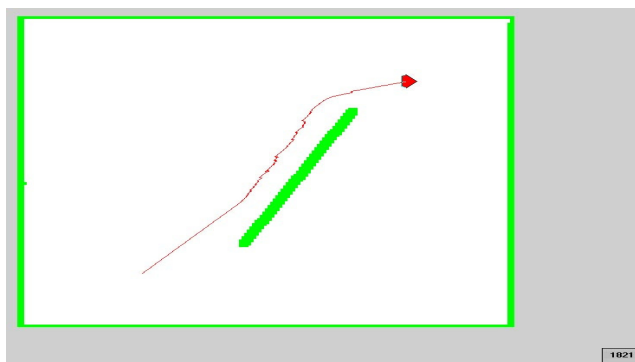
Dôležitú úlohu zohrávajú flagy (horná úroveň riadenia), ktoré sú určené na logické navigovanie robota v krízových situáciách ako sú rohy, prekážky tvaru „U“, nepriechodnosť do cieľa apod. Môže nastať aj situácia, keď ma robot cieľ už za chrbtom (napríklad pri obchádzaní prekážky), vtedy sa robot otáča o uhol tak, aby už v smere cieľa dokázal merať.

## 6. Výsledky simulácií

Najskôr sa testovala metóda lokálneho plánovania pomocou metódy potenciálových polí. Ako vidieť na výsledkoch, bola nedokonalejšia, pretože bola použitá ako lokálna metóda a robot nepozná celý potenciál prekážky. V kombinácii s hľadaním najkratšej nožnej cesty k cieľu tak prichádza k zacykleniu sa.

Tento problém je ošetrený použitím zadného ochranného snímača, čo však vedie na energeticky náročné riešenie. Výhodnejším riešením sa zdala zvolená vlastná modifikácia metódy napnutého vlákna. Táto metóda sa asi najviac približuje k orientácii človeka v neznámom prostredí.

Výsledky obchádzania prekážok lokálneho plánovania cesty pomocou metódy potenciálových polí nedosahovali predpokladané výsledky. Pomocou tejto zvolenej metódy dokázal robot prekážky obchádzať, a aj smerovať do cieľa v rôznych prostrediach, výsledky neboli uspokojivé po ekonomickej a energetickej stránke (Obr. 7, 1821 krokov na dosiahnutie cieľa). V istých prípadoch dochádzalo k zacykleniu, čo nás priviedlo k záveru, že táto metóda nie je vhodná na lokálnu navigáciu mobilného robota v neznámom prostredí.



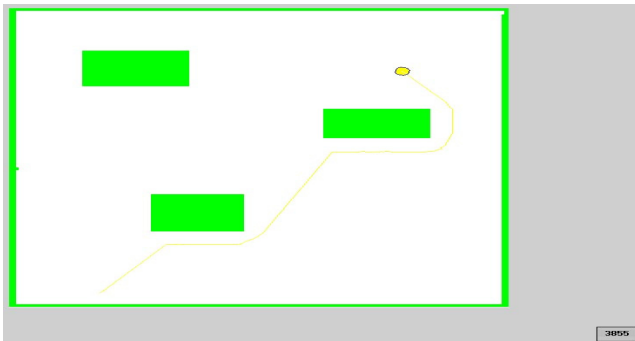
**Obr.7 Lokálna navigácia pomocou metódy potenciálových polí**

**Fig.7 Local navigation with potential field method**

Modifikovaná metóda napnutého vlákna ukazuje dobré výsledky aj po ekonomickej aj energetickej stránke. Na Obr. 8 (3855 krokov na dosiahnutie cieľa) vidieť, že robot obchádza prekážky úsporne a s hladkým priebehom, aj keď dochádza k istým korekciám dôsledkom zaokrúhľovania výpočtov.

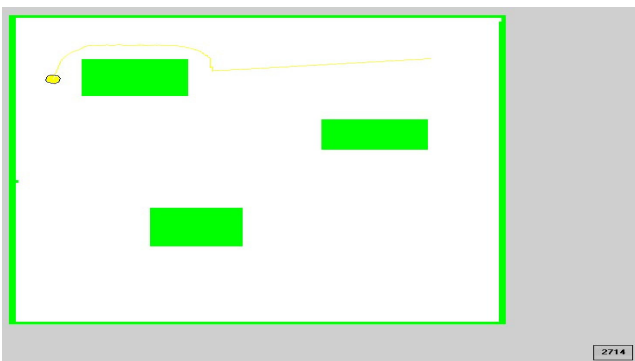
Vidieť, že ide o lokálne plánovanie, pretože obchádzanie nie je optimálne. Optimálnosť vieme zaručiť len využitím globálneho plánovania cesty a trajektórie, kde robot má komplexnejšiu informáciu o tvare a veľkosti prekážok. V prípade zavedenia mobilných prekážok by však bolo nutné vyriešiť možnosť dočasného zablokovania kratšej cesty.



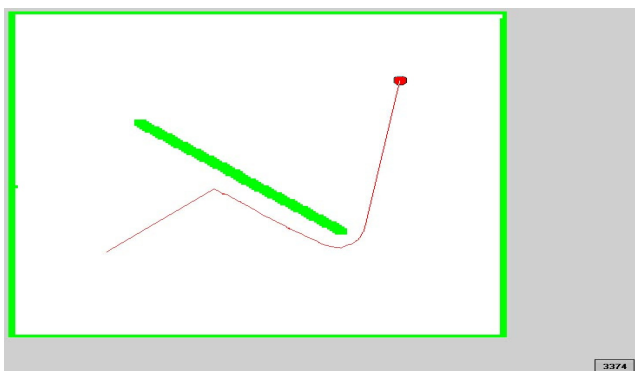


**Obr.8 Metóda napnutého vlákna**  
**Fig.8 Method of tense string**

Na Obr. 9 (2714 krokov na dosiahnutie cieľa) je vidieť problematiku optimálneho obchádzania prekážky, v tomto prípade by sa dalo upraviť obchádzanie prekážky robotom už omnoho skôr, čo však tiež nemusí byť výhodné. Najmä po ekonomickej a energetickej stránke by to nemuselo byť výhodné (veľké oblúky), prípadne by mohlo dôjsť ku konfliktom v algoritme v prípade umiestnenia cieľa do blízkosti prekážky.



**Obr.9 Problematika optimálnosti**  
**Fig.9 Optimality problems**



**Obr.10 Problematika smeru obchádzania**  
**Fig.10 Direction of move about problems**

Obr. 10 (3374 krokov na dosiahnutie cieľa) naznačuje úvahu, či je výhodnejšie obchádzať prekážkou po pravej alebo ľavej strane. Keďže však ide o lokálne plánovanie cesty, robot nemôže rátať s optimálnou cestou.

## Záver

Podľa experimentov môžeme tvrdiť, že bez využitia programových prostriedkov umelej inteligencie, metódy plánovania cesty robota v prostredí so stacionárnymi prekážkami zlyhávajú v istých situáciách, bežných pre život človeka a jeho

orientáciu v prostredí. Náhradou môže byť vhodná kombinácia týchto metód, prípadne ich vlastná modifikácia, najlepšie s rôznymi úrovňami riadenia. Je však zrejmé, že univerzálny dokonalý prostriedok pre orientáciu mobilného robota v akomkoľvek neznámom prostredí nedokážeme v súčasnosti vytvoriť. Takisto môžeme tvrdiť, že pri zabezpečení čo najpresnejšej informácie o polohe, orientácii prípadne aj tvaru prekážok, dokážeme robot naučiť takému správaniu, ktoré síce nie je optimálne, ale postačujúce a rozumné. Preto je nutné zaoberať sa otázkou snímačov, inteligentného vyhodnotenia informácie a integrácie informácie z rôznych senzorických systémov.

## Literatúra

- [1] Jurišica L.: Aplikovaná robotika, Bratislava
- [2] Siegwart R., Nourbakhsh I.R.: Introduction to Autonomous Mobile Robots, Massachusetts Institute of Technology, 2004
- [3] [www.mobilerobotics.org](http://www.mobilerobotics.org)
- [4] dokumentácia Simrobot
- [5] Jurišica L., Pavlovkin J.: Plánovanie dráhy mobilného robota, AT&P Journal 2, 2001
- [6] Jurišica L., Pavlovkin J.: Reprezentácia prostredia mobilného robota, AT&P Journal plus 2, 2001
- [7] Jurišica L., Pavlovkin J.: Navigácia mobilného robota, AT&P Journal 2, 2003
- [8] Jurišica L., Murár R.: Reaktívne riadenie mobilného robota, AT&P Journal 5, 2003
- [9] Jurišica L., Hanzel J.: Tvorba máp neznámeho prostredia pre mobilný robot s využitím ultrazvuku, AT&P Journal 12, 2003
- [10] Jurišica L., Murár R.: Prieskum neznámeho prostredia mobilným robotickým systémom a tvorba topologickej mapy prostredia, AT&P Journal 2 a 3, 2004
- [11] S. Thrun, A. Bücken, Learning Maps for Indoor Mobile Robot Navigation, AI Magazine. 99: 21-71

## Abstract

This paper contents about modelling, control and navigation of mobile robot in unknown structured environment. Results are experimental simulations, which compare two methods of local path planning of mobile robot.

**Ing. František Duchoň**  
**Ing. Roman Murár**

Fakulta elektrotechniky a informatiky  
 Ústav riadenia a priemyselnej informatiky  
 Ilkovičova 3  
 812 19 Bratislava  
 e-mail: [frantisek.duchon@stuba.sk](mailto:frantisek.duchon@stuba.sk)