

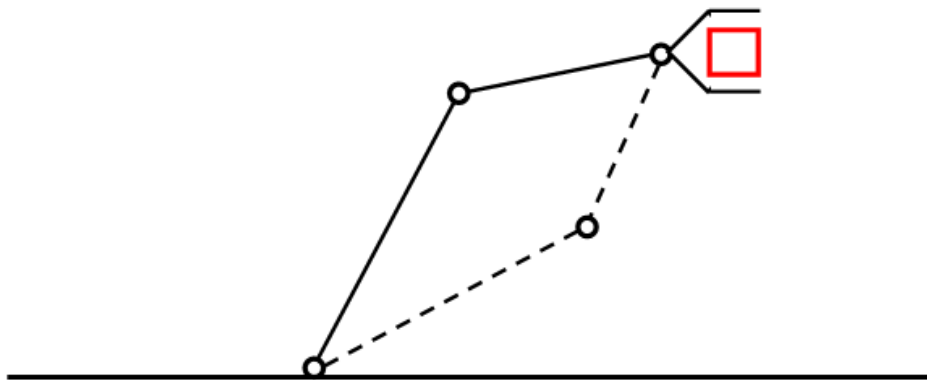
ÚLOHA INVERZNÍ KINEMATIKY – NÁSTROJ PRO ŘÍZENÍ ROBOTICKÉHO MANIPULÁTORU

Jan Crha¹, Miroslav Kiš², Jan Mareš³

ÚVOD

Robotické manipulátory se v poslední době těší stále větší popularitě v nejrůznějších odvětvích průmyslu, v medicíně nebo např. i pyrotechnice. Svědčí o tom celá řada publikací na toto téma z poslední doby, (1-4).

Pro v praxi používané robotické manipulátory je zcela zásadní přesnost a rychlost regulace. Přesná hodnota počítaná dlouhou dobu složitou optimalizační metodou je bezcenná stejně jako rychle vypočtený nepřesný akční zásah. Je tedy nezbytné zvolit „optimální“ přístup, který bude dostatečně dobře a přesně robotické rameno řídit. V principu lze řízení rozdělit do dvou skupin – (i) jednodušší přímá úloha kinematiky, kde je dána pozice všech kloubů ramene a (ii) složitější inverzní úloha kinematiky, kde je dána pouze pozice základny a koncového kloubu. Úkolem regulace manipulátoru je potom nastavení jednotlivých ovládacích servomotorů tak, aby se koncový bod ramene dostal do přesně daného bodu v prostoru. Obecně se v tomto případě jedná o netriviální problém, neboť manipulátory jsou často stavěné s tolika stupni volnosti a daného bodu lze dosáhnout několika možnými nastaveními středového kloubu, viz obrázek 1.



Zdroj: Autor

Obr. 1 – Obecné schéma manipulátoru

Taková úloha lze řešit (v prostoru) v principu několika možnými přístupy. Asi nejkorektnější přístup je výpočet průniku dvou kulových ploch, jejichž poloměry vychází z délek jednotlivých ramen. Výsledkem je kružnice a teoreticky kdekoli na této kružnici se může pozice středového kloubu nacházet. Tento přístup je ale matematicky a programátorsky

¹ Bc. Jan Crha, Vysoká škola chemicko-technologická, Ústav počítačové a řídicí techniky, Technická 5, Praha 6

² Bc. Miroslav Kiš, Vysoká škola chemicko-technologická, Ústav počítačové a řídicí techniky, Technická 5, Praha 6.

³ Ing. Jan Mareš, Ph.D., Vysoká škola chemicko-technologická, Ústav počítačové a řídicí techniky, Technická 5, Praha 6. Tel.:220 444 172, E-mail: jan.mares@vscht.cz

nejednoduchý. V dalším textu jsou ukázány možné přístupy, které jsou prakticky jednoduše využitelné a dávají velmi dobré výsledky.

1. MATEMATICKÝ MODEL RAMENE

Ve skutečném světě se setkáváme s robotickými rameny, na konci kterých je připevněn nástroj. Účelem je umístění tohoto nástroje do cílového bodu. Vysoká přesnost a opakovatelnost navedení je téměř výlučně podmíněna automatickým řízením. Toto vyžaduje vytvoření matematického modelu, který nám poskytne parametry použitelné při vykonávání řídicí úlohy. Tvorbu modelu rozdělíme do následujících kroků:

1. Formulaci všeobecné definice a doplnění o zjednodušující předpoklady.
2. Sestavení systému rovnic popisujících předmět.
3. Doplnění modelu o reálné omezující podmínky.

1.1 Formulace všeobecné definice

Obecnou definici robotického ramene můžeme formulovat následovně:

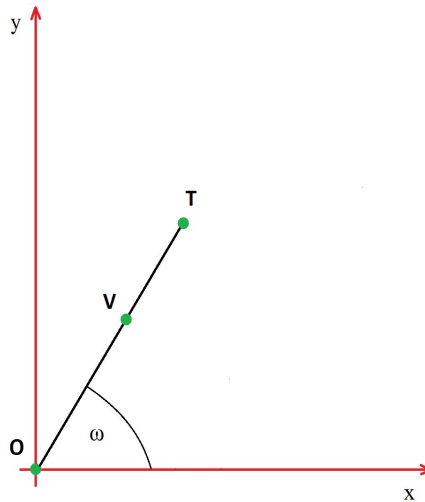
Robotické rameno je manipulační stroj napodobující svou konstrukcí a způsobem fungování lidskou ruku.

Tuto definici je však pro náš případ nutné doplnit a omezit, aby ji bylo možné použít k tvorbě modelu. A to stanovením následujících faktů:

- rameno se skládá z dvou nosníků, definujeme „spodní“ s délkou a a „vrchní“ s délkou b ;
- spodní nosník je připojen kloubem umožňujícím rotaci ve vertikální rovině k podstavě;
- vrchní nosník je připojen kloubem umožňujícím rotaci ve vertikální rovině ke konci spodního nosníku, tento bod nazveme *vrcholem*;
- podstava ramena rotuje v horizontální rovině s osou rotace nacházející se v místě připojení spodního nosníku;
- bod připojení ramene k podstavě ztotožníme s počátkem pravoúhlé souřadnicové soustavy;
- bod opačný k bodu připojení na konci ramene nazveme *koncovým* bodem ramene;

1.2 Sestavení soustavy rovnic

Předpokladem při výběru matematického popisu je základní geometrická konstrukce ramene sestávající ze dvou úseček spojených v jediném bodu označeném *vrchol* ($V = [x_v, y_v, z_v]$) a dále definovaným počátkem souřadnicové soustavy (O) a cílovým bodem systému ($T = [x_t, y_t, z_t]$). Tento systém je definován v třech rozměrech, vzhledem k definici můžeme zkoumat jeho chování nezávisle v horizontální a vertikální rovině. Úhel natočení ramene v horizontální rovině můžeme stanovit ze vztahu (1). Vztahem (2) vyjádříme souřadnici x v kolmém průmětu na vertikální rovinu (x, z) . Situace je načrtnutá na obrázku 2.



Zdroj: Autor

Obr. 2 – Geometrická konstrukce ramene

$$\operatorname{tg}(\omega) = \frac{y_t}{x_t} \Rightarrow \omega = \operatorname{arctg}\left(\frac{y_t}{x_t}\right) \quad (1)$$

$$x' = \sqrt{x^2 + y^2} \quad (2)$$

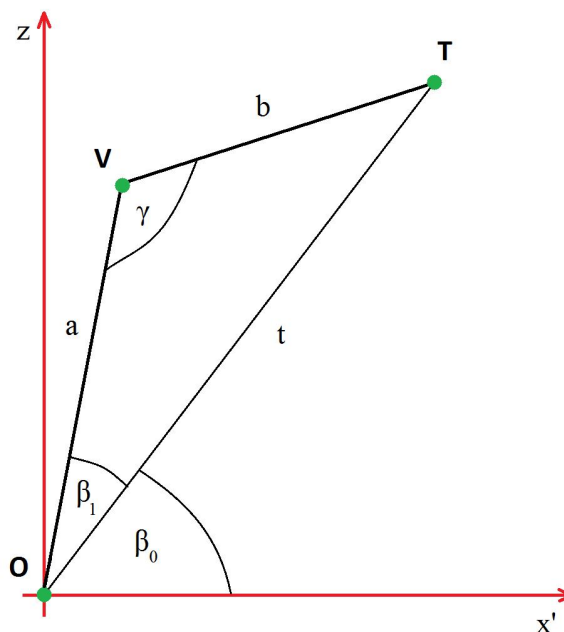
x' – transformovaná souřadnice

x, y – původní souřadnice

Známymi parametry v tomto systému jsou délky úseček, počáteční bod a koncový bod. Neznámé jsou úhel natočení kolmého průmětu v horizontální rovině, úhel definovaný horizontální rovinou a úsečkou [O,V], úhel mezi úsečkami [O,V] a [V,T] a bod V. Na tento systém můžeme nahlížet z dvou různých pohledů.

1.3 Kosinová věta

Jednou možností je sestavit trojúhelník užitím znalosti délky úseček, bodů O, T a faktu, že bod V mají společný, z čehož následně pomocí kosinové věty dopočítáme potřebné vnitřní



úhly. Bez újmy na obecnosti si můžeme zjednodušit výpočet prací jenom ve vertikální rovině určené osou z a body V , T , přičemž souřadnici x transformujeme dle (2). Pro jednotlivé úhly dle obrázku 3 platí vztahy (3).

Zdroj: Autor

Obr. 3 – Využití kosinové věty

$$\begin{aligned}\cos(\gamma) &= \frac{a^2+b^2-t^2}{2ab} \\ \cos(\beta_1) &= \frac{a^2+t^2-b^2}{2at} \\ \operatorname{tg}(\beta_0) &= \frac{z_t}{x'_t}\end{aligned}\tag{3}$$

1.4 Průnik kružnic

Další možností je určení vektorů $[O,V]$ a $[V,T]$, kde bod V najdeme jako průnik kružnic $k_1(O, |[O,V]|)$ a $k_2(T, |[V,T]|)$, z čeho následně spočítáme hledané úhly. Tato situace je načrtnutá na obrázku 3. Při použití tohoto postupu uvažujeme rovnice kružnic ve tvaru (4), bez újmy na obecnosti můžeme jednu kružnici umístit do počátku a druhou do obecného

$$\begin{aligned}k_1: x^2 + y^2 &= r_1^2 \\ k_2: (x - x_0)^2 + (y - y_0)^2 &= r_2^2\end{aligned}\tag{4}$$

bodu.

Z rovnic pro dvě kružnice vytvoříme soustavu. Počet řešení této soustavy závisí na vzdálenosti středů (R). Za předpokladu konstantních poloměrů (r_1 , r_2), což zodpovídá konstantní délce ramen, mohou nastat následovní situace:

- a. Žádné řešení: $R > r_1 + r_2$ v $R < |r_1 - r_2|$
- b. Právě jedno řešení: $R = r_1 + r_2$ v $R = |r_1 - r_2|$
- c. Právě dvě řešení: $R \in (|r_1 - r_2|, r_1 + r_2)$

Základním problémem tohoto postupu je prvotní nejednoznačnost řešení. Vzhledem k tomu, že pracujeme s kvadratickými rovnicemi, tak počet řešení může být větší než jedno a je nutné využít poznatky o systému na eliminaci nevyhovujících řešení. V principu můžeme formulovat podmínky alespoň dvou druhů – optimalizační a konstrukční.

Další nepříznivou vlastností tohoto postupu je výpočetní náročnost – řešíme soustavu dvou kvadratických rovnic o dvou neznámých. Za předpokladu existence průniku kružnice k_1 (5), k_2 (6) a kladných délek ramen, můžeme určit body průniku ze vztahů (7), z čeho dostaneme souřadnice bodů V_1 a V_2 dle (8). Pracujeme v rovině a souřadnici x transformujeme dle (2).

$$a \equiv [O, V]; b \equiv [V, T]$$

$$k_1: b^2 = (x' - x_t)^2 + (z - z_t)^2 \quad (5)$$

$$k_2: a^2 = x'^2 + z^2 \quad (6)$$

$$x'_1 = \frac{x_t}{2} + \left(x_t \left(\frac{a^2}{2} - \frac{b^2}{2} \right) - \frac{1}{2} (x_t \sqrt{w}) \right) (x_t^2 + z_t^2)^{-1}$$

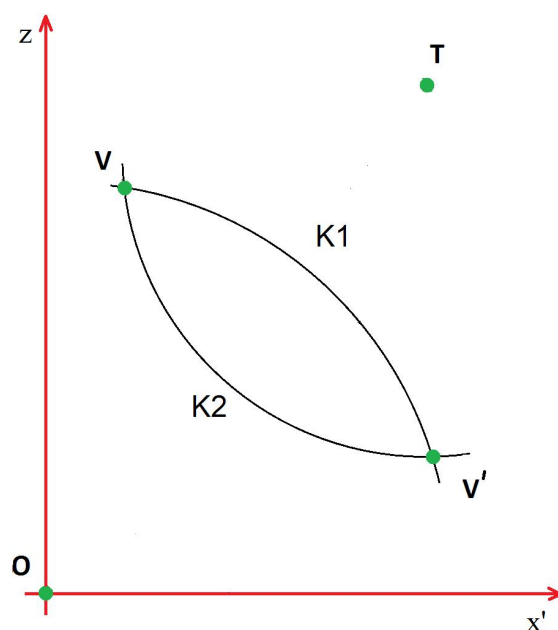
$$z_1 = \frac{z_t}{2} + \left(z_t \left(\frac{a^2}{2} - \frac{b^2}{2} \right) - \frac{1}{2} (z_t \sqrt{w}) \right) (x_t^2 + z_t^2)^{-1}$$

$$x'_2 = \frac{x_t}{2} + \left(x_t \left(\frac{a^2}{2} - \frac{b^2}{2} \right) - \frac{1}{2} (x_t \sqrt{w}) \right) (x_t^2 + z_t^2)^{-1} \quad (7)$$

$$z_2 = \frac{z_t}{2} - \left(z_t \left(\frac{a^2}{2} - \frac{b^2}{2} \right) - \frac{1}{2} (z_t \sqrt{w}) \right) (x_t^2 + z_t^2)^{-1}$$

$$w = (-a^2 + 2ab - b^2 + x_t^2 + z_t^2)(a^2 + 2ab + b^2 - x_t^2 - z_t^2)$$

$$V_1 = [x'_1, z_1]; V_2 = [x'_2, z_2] \quad (8)$$



Zdroj: Autor

Obr. 4 – Využití průniku kružnic

1.5 Doplnění o reálná omezení

Vytvořený model musí být v reálném použití konfrontován s omezujícími podmínkami reálného světa. Některé podmínky je dobré řešit před spočítáním hlavního modelu. Jedná se o podmínky dosažitelnosti, které je vhodné stanovit formou definičního oboru. Dalšími jsou podmínky optimalizační, které je vhodné naopak řešit až po spočítání modelu výběrem z možných variant.

2. APLIKAČNÍ PŘÍKLAD

Úloha přímé a inverzní kinematiky je demonstrována na laboratorním modelu robotického manipulátoru, který byl vytvořen na Ústavu počítačové a řídicí techniky VŠCHT V Praze, více viz (5).

2.1 Popis reálného systému

Model je postaven ze dvou sad lego Mindstorms NXT, což umožnilo použít šest motorů, viz obr. 5. Základnu tvoří dvě NXT kostky a jeden motor umístěný mezi nimi. Tento motor otáčí celým ramenem a jeho pohyb je limitován pouze vodiči, které vedou k dalším motorům uchyceným na rameni. Na točně tohoto motoru je připevněna velká převodovka, která zvedá spodní nosník ramene o celkové délce 23cm. Přibližně půlku délky tohoto nosníku tvoří další tři motory a kvůli jejich vysoké váze musela být v průběhu stavby převodovka několikrát zesilována a posléze musel být přidán ještě jeden motor.

Umístění tří motorů na konec spodního nosníku bylo umožněno jen díky homokinetickým kloubům. Pokud by je stavebnice neobsahovala, musely by být dva motory umístěny až u zápěstí, což by vedlo k ještě většímu zatížení motorů zvedajících celé rameno a nejspíše i k nemožnosti dokončit rameno pouze se stavebnicí LEGO. Motory na konci spodního nosníku obstarávají zvedání vrchního nosníku, otáčení zápěstí a úchop čelistí. Vše je vyřešeno důmyslně tak, že je možné vrchní nosník sklopit téměř k spodnímu nosníku nebo ho narovnat tak, aby byl souběžný se spodním nosníkem. Délka horního nosníku je 10cm a zápěstí se zavřenými čelistmi je dlouhé dalších 10cm.

Zapojení motorů do kostek není náhodné, protože každá kostka může pohybovat pouze jedním motorem. Výjimku tvoří dva motory zvedající celé rameno, ty jsou softwarově spojené do jednoho motoru a fyzicky jsou spojeny hřídelí. Ostatní motory jsou zapojeny tak, aby ty, které se mohou nebo musí pohybovat společně, byly zapojeny do různých kostek. Do jedné kostky jsou zapojeny motory hlavní převodovky a motor, který obsluhuje čelisti. Druhá kostka má připojené motory pro rotaci celého ramene, rotaci zápěstí a zvedání koncového nosníku. Motory pro zápěstí a úchop běží vždy spolu a motory pro zvedání ramene a motor pro zvedání nosníku také běží současně.



Zdroj: Autor

Obr. 5 – Model robotického manipulátoru

Rameno bylo postaveno s pěti stupni volnosti – horizontální rotace v základně, zvedání celého ramene, sklápění koncového nosníku, rotace v zápěstí a úchop. Jelikož rotace zápěstí a úchop závisí na objektu/materiálu, který rameno přenáší, dále se jimi zabývat nebudeme a budeme testovat výše zmíněné metody polohování a navádění do žádané pozice.

2.2 Software pro pohyb ramene

Model je ovládán z počítače s využitím programového prostředí MATLAB. NXT kostky umožňují připojení pomocí USB, nebo Bluetooth technologie, která je ale pomalejší a méně spolehlivá. Obě kostky jsou připojeny pomocí dvou USB kabelů (vzhledem k tomu, že je model statický, plně to vyhovuje našim požadavkům).

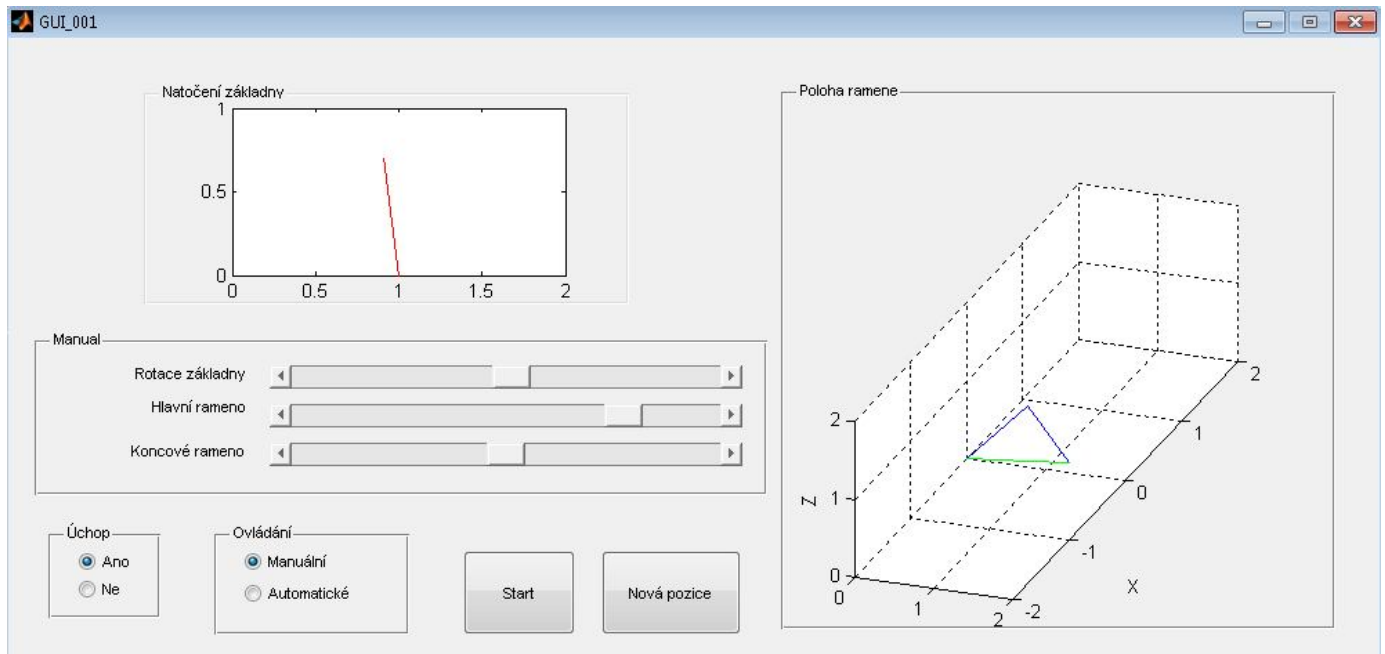
Pro přehlednější ovládání ramene bylo již dříve vytvořeno grafické uživatelské rozhraní, které posloužilo jako základ této úlohy, viz (6). Uživatelské rozhraní bylo modifikováno tak, aby umožňovalo realizaci obou úloh – původní přímý způsob ovládání jednotlivých motorů a nový inverzní způsob ovládání.

2.2.1 Přímá úloha kinematiky

Tento způsob řízení umožňuje každému motoru samostatný pohyb nezávisle na ostatních, což je programově jednoduché, ale ovládání konce ramene se tím velmi zesložituje.

Pokud se pohne libovolným motorem, konec ramene se pohne vždy alespoň ve dvou osách.

Uživatelské rozhraní je upraveno tak, aby všechny posuvníky byly nastaveny přesně na rozsah jednotlivých motorů, a pro lepší orientaci obsluhy je zobrazován i 3D graf s aktuální polohou ramene (včetně průmětu ramene do základny), viz obr 6.

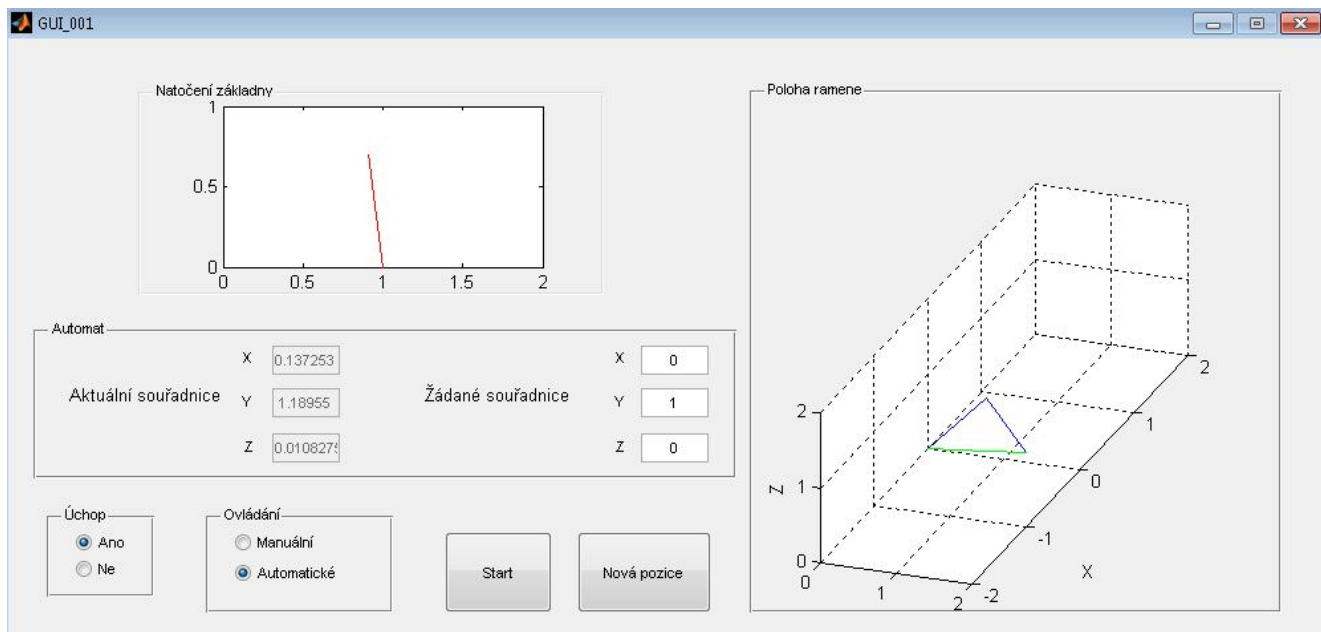


Zdroj: Autor

Obr. 6 – Uživatelské rozhraní - ovládání

2.2.2 Inverzní metoda

Druhá metoda využívá řešení pomocí Kosinové věty. V uživatelském rozhraní se přepnutí ovládání provede v boxu *Ovládání* vybráním možnosti „Automatické“. To způsobí, že se změní ovládací panel s posuvníky - objeví se šest boxů (pro každou souřadnici zvlášť). Jedna trojice souřadnic ukazuje aktuální polohu konce ramene a do druhé trojice souřadnic se zapisuje nová požadovaná poloha ramene.



Zdroj: Autor

Obr. 7 – Uživatelské rozhraní – automatické řízení

Při jakékoli změně údajů v libovolném poli se spustí funkce, která nejprve zkontroluje, zda jsou zadána čísla. Pokud uživatel zapíše jiné znaky než čísla, upozorní ho výstražné okno na danou chybu a hodnota se automaticky nastaví na nulu. Je-li zadané číslo, vypočítá potřebné nastavení motorů, je-li žádaná poloha nedosažitelná, upozorní to uživatele výstražným oknem a hodnota se opět nastaví na nulu.

Následně se provede vlastní výpočet dle kap.1.3, který probíhá ve dvou funkcích, první funkce `arm_angle(target, a, b, method)` počítá úhel, který musí svírat jednotlivá ramena, aby byl koncový bod v žádané pozici. Její argumenty jsou: souřadnice bodu ve formě vektoru `target`, délky jednotlivých nosníků jako argumenty `a` a `b`, a poslední argument `method` slouží k vybrání metody počítání, pro kosinovou větu je to parametr „*cos*“ a pro průnik kružnic parametr „*point*“.

Druhá funkce `motor_angle(ang2, GP, RPO)` přepočítává úhly v radiánech na úhly ve stupních, o které se musí otočit hřídele jednotlivých motorů. Její vstupní parametry jsou `ang2`, což je vektor obsahující požadované nastavení motorů. Objekty `GP` a `RPO` slouží pro ukládání specifických nastavení motorů, jako je například stupně o kolik se mají natočit hřídele motorů, síla jakou se mají točit, směr otáčení a jejich poloha.

Funkce se vždy spustí při změně žádaných souřadnic, konkrétně souřadnice Z.

```

% Vystup:
% omega[rad] - otočení základny, základna úhlu je osa x
% beta[rad] - zdvih ramene "a", se základnou v rovině x,z
% gamma[rad] - úhel mezi ramenem "a" a ramenem "b", v rovině x,z

function [omega,beta,gamma] = arm_angle(target,a,b,method)

xt = target(1);
yt = target(2);
zt = target(3);
fak = 1;%360/2/pi;
c = sqrt(xt^2+yt^2+zt^2);

if(c < a+b && c > a-b )
    omega =(fak*atan(yt/xt));
    if(strcmp('cos',method))
        beta = fak*(acos((a^2+c^2-b^2)/(2*a*c)) + atan(zt/norm([xt,yt])));
        gamma = fak*acos((b^2+a^2-c^2)/(2*a*b));
    elseif ...
    ...
else sprintf('Unknown method');
    omega = -1;
    beta = -1;
    gamma = -1;
end
else
    omega = -1;
    beta = -1;
    gamma = -1;
end
end

```

Vybraná část kódu funkce arm_angle.m

```

function [GP, RPO]=motor_angle(ang,GP,RPO)
% natočení v základně
alf=(ang(1)/pi*765);
GP.scr_L=floor(abs(RPO.scr_pos_set-alf));
if(RPO.scr_pos_set<alf)
    GP.scr_P=15; %otáčení ->
    GP.scr_pos_set=RPO.scr_pos_set+GP.scr_L;
else
    GP.scr_P=-15; %otáčení <-
    GP.scr_pos_set=RPO.scr_pos_set-GP.scr_L;
end
...
end

```

Část funkce motor_angle.m

```

function Set_Z_Callback(hObject, eventdata, handles)

global ang2 pos GP RPO
pos(3) = str2double(get(hObject, 'String'));
if isnan(pos(3))
    set(hObject, 'String', 0);
    pos(3)=0;
    errordlg('Vstup musí být číslo','Error');
else
    [ang2(1), ang2(2), ang2(3)]=arm_angle(pos,1,0.85,'point');
    if (ang2==[-1, -1, -1])
        set(hObject, 'String', 0);
        errordlg('Žádaná pozice je nedosažitelná','Error');
    else
        [GP,RPO]=motor_angle(ang2,GP,RPO);
    end
end
end

```

3. ZÁVĚR

Příspěvek představuje matematické pozadí automatického řízení robotického manipulátoru. První část se zabývá modelem a řešením úlohy inverzní kinematiky s využitím analytické geometrie, (i) průniku dvou kružnic a (ii) Kosinové věty. Z hlediska aplikace je výrazně vhodnější druhá varianta, která má nízké výpočetní nároky a pouze jediné řešení.

Tato varianta byla poté aplikována ve druhé části v případové studii, kde byl řízen zmenšený model robotického manipulátoru (vše s využitím programového prostředí MATLAB). Pro lepší srozumitelnost je text doplněn o výpisy kódu.

POUŽITÁ LITERATURA

- (1) DOLEŽEL, P., HAVLÍČEK, L., MAREŠ, J. Piecewise-Linear Neural Model for Helicopter Elevation Control. *International Journal of Control Science and Engineering* vol. 2, no. 3. 2012. pp.42-46. ISSN 2168-4952..
- (2) DING, J.N., GOLDMAN, R.E., XU, K., ALLEN, P.K., FOWLER, D.L., SIMAAN, N. Design and Coordination Kinematics of an Insertable Robotic Effectors Platform for Single-Port Access Surgery *IEEE-ASME TRANSACTIONS ON MECHATRONICS*. Vol. 18, no. 5. 2013.
- (3) TREESATAYAPUN, C. Discrete-time direct adaptive control for robotic systems based on model-free and if-then rules operation. *INTERNATIONAL JOURNAL OF ADVANCED MANUFACTURING TECHNOLOGY* vol. 68 no. 1. 2013
- (4) MARTINEZ-PRADO, M., FRANCO-GASCA, A., HERRERA-RUIZ, G., SOTO-DORNTES, O. Multi-axis motion controller for robotic applications implemented on an FPGA. *INTERNATIONAL JOURNAL OF ADVANCED MANUFACTURING TECHNOLOGY* vol. 67, no. 9. 2013
- (5) MAREŠ, J., GRÁFOVÁ, L., PROCHÁZKA, A. On-line sledování pohybujícího se předmětu s využitím digitální kamery. *Perner's Contact*, vol. 6, no 1, april 2011. ISSN 1801-674X.
- (6) CRHA, J., MAREŠ, J. Model of Industrial Robotic Manipulator. *In proceedings of 10th International Conference Process Control 2012*. Pp. C012b-1 - C021b-5. Pardubice 2012. ISBN 978-80-7395-500-7.