

## INVERSE KINEMATIC MODELLING IN SIMMECHANICS AND APPLICATION FOR ROBOTIC MECHANISMS

R. Grepl<sup>1</sup>, Č. Ondrůšek<sup>2</sup>

**Summary:** *This paper deals with the design of inverse kinematic model in software MATLAB–SIMMECHANICS. Presented approach is applied to particular robotic mechanism, the panthographical leg of quadruped walking robot. It is possible to use created model as an approximation method training data generator or directly in computation in the MATLAB.*

### 1. Úvod

V poslední době se v rámci řešení projektů AV ČR a dalších výzkumných prací zabýváme modelováním kinematiky a dynamiky mechanismů, zejména robotických pedových soustav. Přímé i inverzní kinematické modelování kráčejících mechanismů je nezbytnou součástí návrhu a optimalizace konstrukce i vlastního řízení, přičemž v jednotlivých etapách konstrukční práce jsou na vytvořené modely kladeny různé požadavky. Obecně se jedná o poměrně obtížný problém, často analyticky neřešitelný. Z hlediska návrhu a testování řídicích algoritmů a optimalizace konstrukce je podstatné, aby navržený kinematický model byl snadno integrovatelný do ostatních konstrukčních nástrojů (simulační software, konstrukční software, simulace řízení), aby jeho změny bylo možno provádět velmi rychle a pokud možno pracoval se zaručenou malou vlastní numerickou chybou. Z hlediska implementace řízení do „malých“ procesorů fyzikálního modelu je pak podstatná rychlost výpočtu modelu a paměťové nároky.

V příspěvku [1] jsme popsali problematiku tvorby numerických inverzních kinematických modelů pomocí metody Lazy Learning. Uvedený přístup funguje velmi dobře, vyžaduje však časově poměrně náročnou fázi „ladění“ parametrů metody a nutnost odvození přímého kinematického modelu (což může být u složitého kinematického mechanismu pracné a může být zdrojem lidských chyb). Omezující je také samotné použití Lazy Learning Toolboxu, který je určen pouze pro nekomerční a nevojenské aplikace. Výhodou je naopak vysoká rychlost výpočtu. Z tohoto pohledu se přístup jeví vhodný pro střední až finální část konstrukční práce.

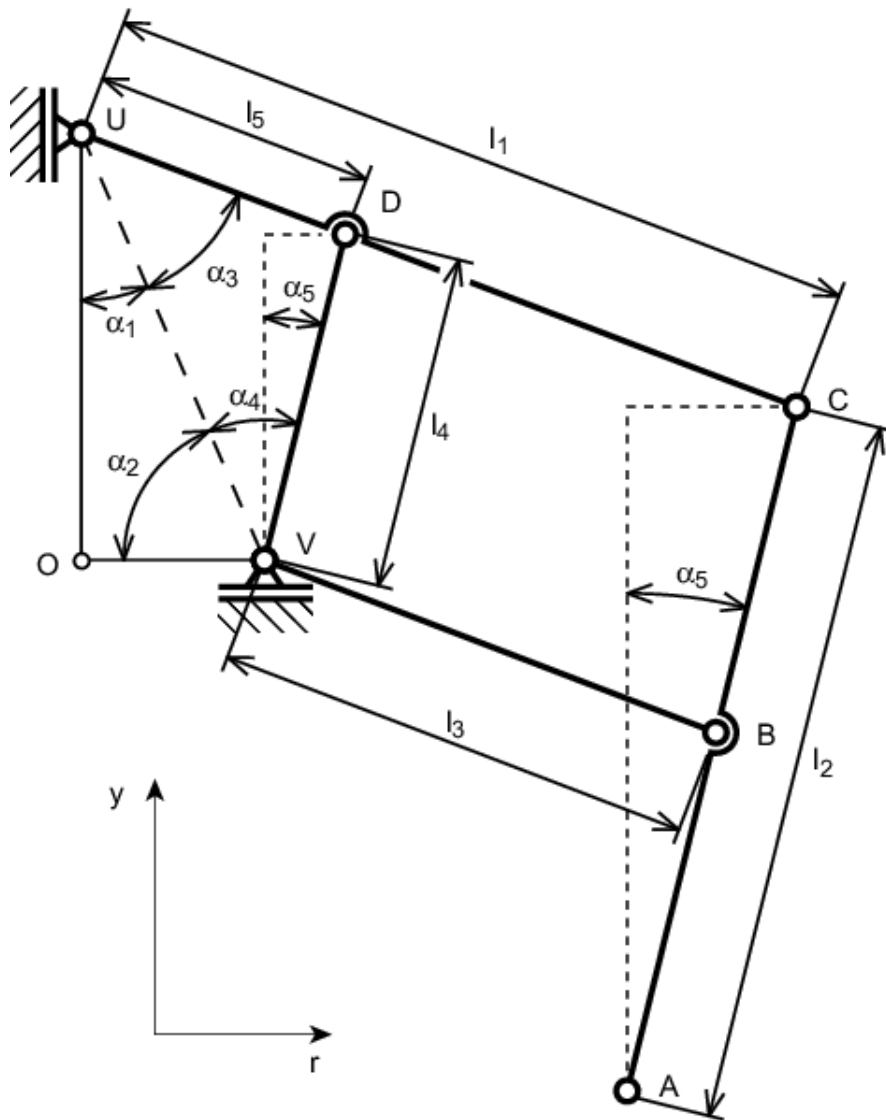
<sup>1</sup>Ing. Robert Grepl, Laboratoř mechatroniky a robotiky, Ústav termomechaniky Akademie věd ČR  
Technická 2, 616 69, Brno, tel.: +420 541 142 757, e-mail: grepl@feec.vutbr.cz

<sup>2</sup>Doc.Ing. Čestmír Ondrůšek, CSc., ÚVEE FEKT VUT v Brně, Technická 8, 616 69, Brno; 5 4114 2364,  
e-mail: ondrusek@feec.vutbr.cz

V tomto článku bychom rádi prezentovali další postup naší práce, který se soustředil na vytvoření metody návrhu inverzního kinematického modelu přímo v prostředí simulačního software Matlab - SimMechanics bez použití dalších nástrojů a bez nutnosti odvozovat analytický přímý model kinematiky mechanismu. Za cenu mírného snížení rychlosti výpočtu modelu dojde k podstatnému zkrácení doby jeho návrhu, což představuje v počáteční a střední části konstrukčních prací významnou výhodu. Vytvořený model lze použít přímo v simulačním výpočtu nebo jako generátor dat pro libovolný aproximátor (neuronová síť, neurofuzzy, Lazy Learning, LWR apod.).

## 2. Formulace problému

Pro realizovaný fyzikální model byla zvolena koncepce pantografické nohy se třemi stupni volnosti, schéma je na obr. 1, realizace pak na obr. 2. Z hlediska numerického řešení simulačních

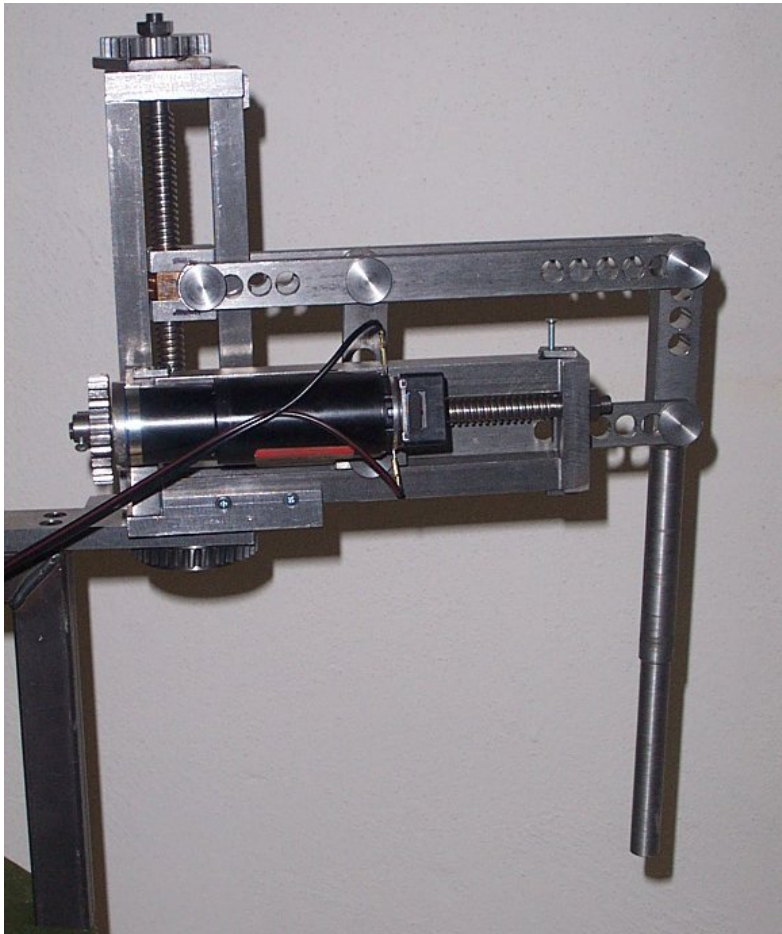


Obr. 1: Geometrie pantografické robotické nohy

výpočtů je důležitý fakt, že se jedná o uzavřený kinematický řetězec, což představuje (na vnitřní

úrovni výpočtu) komplikaci, jejímž výsledkem je zvýšení výpočetní náročnosti, případně nutnost citlivého nastavení řešičů vazeb (constraints).

Obecně lze analytický kinematický model chápat jako soustavu algebraických rovnic transformující jeden souřadný systém určující polohu mechanismu na jiný. V případě, že z nějakého důvodu nelze algebraické rovnice sestavit, musíme transformaci provést jinou (numerickou) metodou. V našem případě je jedním souřadným systémem poloha koncového bodu  $A$  definovaná cylindrickými souřadnicemi  $r$  a  $y$  a druhým pak polohy motorů  $U$  a  $V$ . Poznamenejme,



Obr. 2: Fyzikální model robotické nohy

že vzhledem k tomu, že z hlediska kinematického řízení je otáčení kolem svislé osy  $y$  triviální záležitostí, řešíme kinematický model pouze ve svislé rovině  $r - y$  otáčející se kolem osy  $z$ . Přímý kinematický model (PKM) má tvar:

$$[U, V] \rightarrow [r_A, y_A] \quad (1)$$

Tento přímý model je pro náš případ pantografické robotické nohy snadno řešitelný a vyjádřitelný v podobě analytického modelu. Oproti tomu inverzní kinematický model (IKM)

$$[r_A, y_A] \rightarrow [U, V] \quad (2)$$

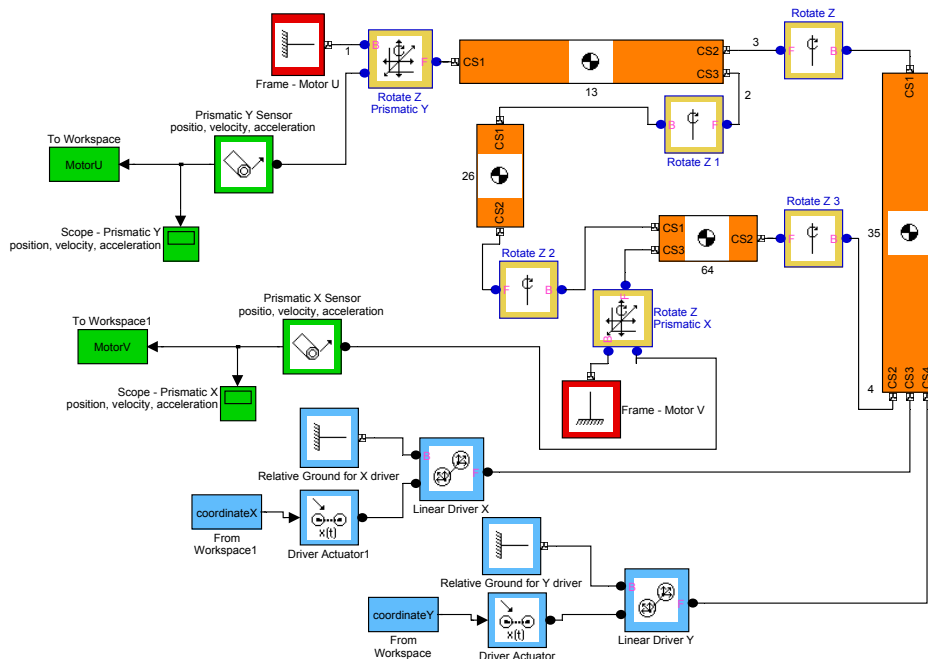
analyticky řešitelný není. Při standardním použití numerického aproximátoru (např. [1]) převedeme úlohu IKM na mnohonásobné opakování PKM a výsledky IKM pak získáme aproximací.

### 3. Postup řešení

Při využití programu SIMMECHANICS je možné provést výpočet IKM přímo, bez nutnosti sestavovat složitým a časově náročným způsobem analytický PKM.

Na obr. 3 je znázorněno schéma modelu v SIMULINK – SIMMECHANICS. Oranžové, žluté a červené bloky reprezentují samotnou mechanickou část modelu. Jsou to bloky `Body`, `Joint` a `Ground`. Takto je sestaven mechanismus bez ohledu na to, jaká bude konkrétně úloha, kterou chceme řešit (IKM, dynamický model, přímé nebo inverzní modely).

Modrá část schématu reprezentuje bloky nutné pro vstupní část algoritmu IKM. Pomocí `Linear Driver` a `Driver Actuator` definujeme trajektorii koncového členu (bodu *A*) mechanismu (poloha, rychlost, zrychlení). Obecně lze nadefinovat libovolnou přechodovou křivku (příčemž je vhodné ji znát v analytickém tvaru a vyhnout se tak nutnosti numerické derivace). V nejjednodušším případě, kdy použijeme IKM jako generátor dat pro aproximační metodu nebo kdy potřebujeme znát hodnoty výstupů z IKM v jednotlivých bodech, je předepsaná trajektorie reprezentována hodnotou polohy, rychlosti a zrychlení v jednom bodě.



Obr. 3: *Inverse kinematic model in SimMechanics*

Po spuštění simulace se SIMMECHANICS snaží zaujmout předepsanou polohu. V případě možné mnohoznačnosti pak vybírá polohu nejbližší počátečnímu stavu (důležité při návrhu počáteční konfigurace modelu). V případě, že je požadovaná poloha nerealizovatelná, řešič vazeb (constraints) ohlásí chybu, přičemž je k dispozici několik různých možných nastavení řešiče a jeho tolerancí.

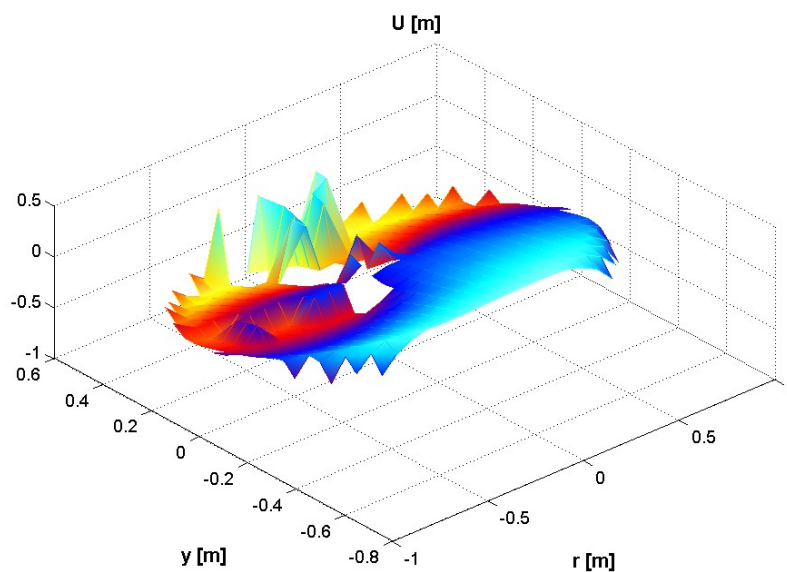
Zelená část schématu obsahuje bloky nutné pro výstupní část IKM. Blok `Joint Sensor` odečítá zvolené veličiny z příslušného `Joint` a tyto jsou uloženy do `Workspace`.

#### 4. Výsledky

Uvedený postup sestavení IKM v SIMMECHANICS jsme ověřili simulačními výpočty pro konkrétní rozměry nohy robotu. Zvolený rozsah pohybu mechanismu  $r \times y$  byl rovnoměrně rozdělen mřížkou a v jejích uzlových bodech byly vypočítány hodnoty  $U$  a  $V$ . Parametry výpočtu jsou uvedeny v tab. 1. Na obr. 4 a 5 jsou vykresleny vypočtené funkce polohy motoru  $U$  a  $V$ . Na

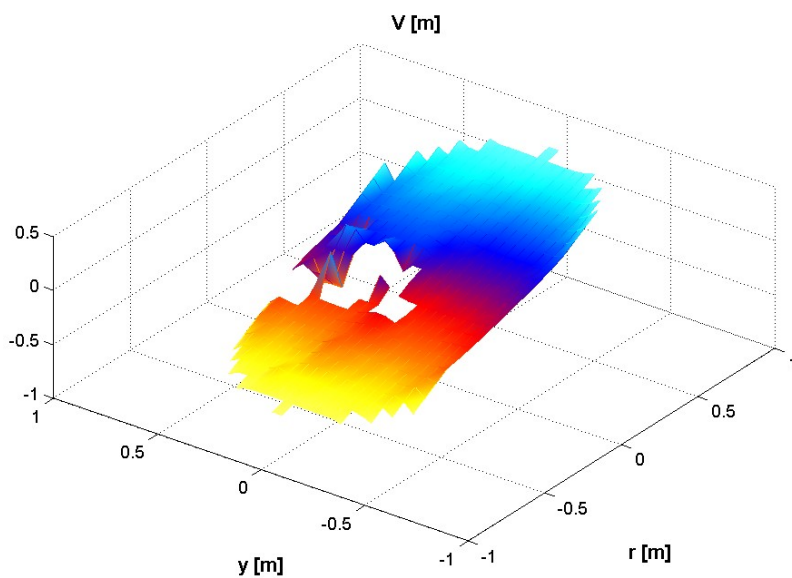
$L_1$	400mm
$L_2$	500mm
$L_3$	250mm
$L_4$	270mm
rozsah $r$	$-900 \div 900mm$
rozsah $y$	$-530 \div 520mm$
matice výpočtu $n_r \times n_y$	$25 \times 24$

Tab. 1: Parametry výpočtu IKM

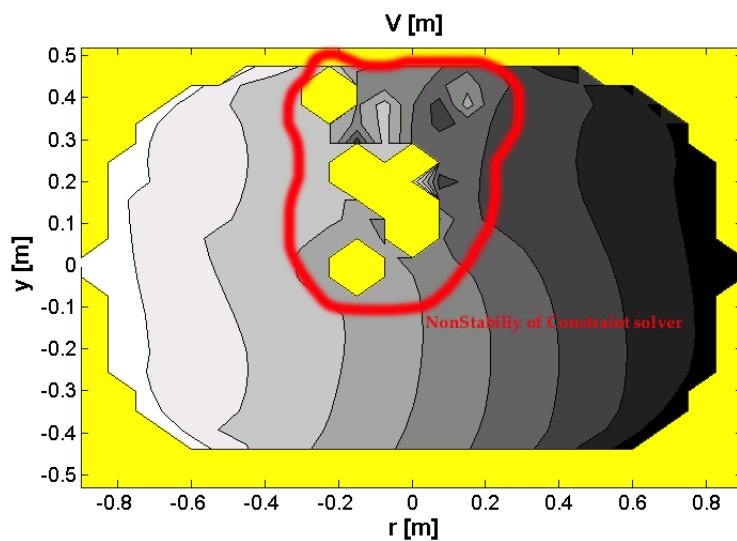


Obr. 4: Vypočtené polohy motoru  $U$

obrázku 6 je dobře patrný tvar pohybového prostoru nohy (žlutou barvou je vyznačena oblast nedosažitelnosti) a také oblast nestability nastavení počátečních podmínek řešiče vazeb, která je vidět i na charakteru funkcí  $U$  a  $V$ .



Obr. 5: Vypočtené polohy motoru  $V$



Obr. 6: Vyznačení kinematicky možných poloh mechanismu a oblasti nestability řešiče vazeb

## 5. Závěr

Simulační výpočty ukazují, že v příspěvku popsaná metoda tvorby IKM v prostředí SimMechanics je velmi dobře použitelná. Při implementaci IKM do dalších výpočtů (např. dynamický model, komplexní model) je nutno vyloučit z pracovního prostoru mechanismu oblast nestability řešiče vazeb (vyznačeno na obr. 6) nebo v této oblasti provést jemnější dělení mřížky a speciální nastavení počáteční polohy mechanismu tak, aby řešič vazeb pracoval korektně. V ostatních částech pracovního prostoru je přesnost na úrovni nastavených tolerancí řešiče vazeb, tedy velmi dobrá. Rychlost výpočtu (simulace) je pod jednu sekundu, v případě, že požadujeme odpověď IKM v bodě, který je pro mechanismus nerealizovatelný (žlutá oblast na obr. 6), doba výpočtu je mírně větší.

## Poděkování

Práce vznikla za podpory pilotního projektu ÚT AV ČR č. 52020 "Řízení kráčivého robotu s využitím metod umělé inteligence", následujícího projektu "Realizace základních řídicích členů kráčivého robotu", projektu MŠMT MSM 262100024 "Výzkum a vývoj mechatronických soustav" a výzkumného záměru CEZ: J22/98: 261100009 "Netradiční metody studia komplexních a neurčitých systémů".

## Reference

- [1] Grepl, R., Kratochvíl, C.: *Inverzní kinematický model robotické nohy metodou Lazy Learning*, Výpočtová mechanika 2002, 29.-31. říjen 2002, Nečtiny, Česká republika,
- [2] Houška, P., Březina, T., Singule, V.: *Koncepce mechatronické soustavy nohy kráčivého robotu*, Inženýrská mechanika 2002, Svratka, Česká republika,
- [3] Miček P.: *Omezení vlivu singularit při posuzování neuro-fuzzy modelu*, Inženýrská mechanika 2002, Svratka, Česká republika,