



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**  
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA STROJNÍHO INŽENÝRSTVÍ**  
**ÚSTAV MECHANIKY TĚLES, MECHATRONIKY A**  
**BIOMECHANIKY**

FACULTY OF MECHANICAL ENGINEERING  
INSTITUTE OF SOLID MECHANICS, MECHATRONICS AND  
BIOMECHANIC

**ANALÝZA KOMPLEXNÍHO NÁVRHU**  
**PRŮMYSLOVÉHO MANIPULÁTORU**  
COMPLEX DESIGN OF INDUSTRIAL MANIPULATOR ANALYSIS

**BAKALÁŘSKÁ PRÁCE**  
BACHELOR'S THESIS

**AUTOR PRÁCE**  
AUTHOR

**LUBOŠ PEML**

**VEDOUCÍ PRÁCE**  
SUPERVISOR

**Ing. LUKÁŠ BŘEZINA, Ph.D.**

BRNO 2012

Vysoké učení technické v Brně, Fakulta strojního inženýrství

Ústav mechaniky těles, mechatroniky a biomechaniky

Akademický rok: 2011/2012

## **ZADÁNÍ BAKALÁŘSKÉ PRÁCE**

student(ka): Luboš Pemeš

který/která studuje v **bakalářském studijním programu**

obor: **Mechatronika (3906R001)**

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma bakalářské práce:

### **Analýza komplexního návrhu průmyslového manipulátoru**

V anglickém jazyce:

#### **Complex design of industrial manipulator analysis**

Stručná charakteristika problematiky úkolu:

Robotický manipulátor je v současné době v různých modifikacích nepostradatelnou součástí průmyslových aplikací. Úkolem této práce je na základě dostupné literatury analyzovat a popsat stěžejní kroky při návrhu průmyslového manipulátoru a tyto demonstrovat na jednoduchém příkladu manipulátoru.

Cíle bakalářské práce:

- 1) Analyzujte postup pro komplexní návrh průmyslového manipulátoru.
- 2) Demonstrujte stěžejní body postupu na jednoduchém příkladu rovinného manipulátoru.

Seznam odborné literatury:

Jazar, R.N.: Theory of applied robotics, Springer, 2007

Siciliano, B. : Robotics Modeling, planning and control, Springer, 2010

Spong, M.W. : Robot modeling and control, John Wiley and Sons, 2007

Vedoucí bakalářské práce: Ing. Lukáš Březina, Ph.D.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2011/2012.

V Brně, dne 9.11.2011

L. S.

---

prof. Ing. Jindřich Petruška, CSc.  
Ředitel ústavu

---

prof. RNDr. Miroslav Doupovec, CSc.  
Děkan fakulty

**Abstrakt:**

Práce se zabývá analýzou postupu pro komplexní návrh průmyslového manipulátoru a demonstrací hlavních bodů postupu na příkladu rovinného manipulátoru s využitím programů Simulink a MSC Adams.

**Abstract:**

The work deals with analysis of industrial manipulator complex design procedure and demonstrates major issues of that procedure on planar manipulator example using Simulink and MSC Adams.

**Klíčová slova:**

Průmyslový robot, manipulátor, rovinný manipulátor, kinematika.

**Keywords:**

Industrial robot, manipulator, planar manipulator, kinematics.

**Bibliografická citace:**

PEML, L. *Analýza komplexního návrhu průmyslového manipulátoru*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2012. 47 s. Vedoucí bakalářské práce Ing. Lukáš Březina, Ph.D.

**Poděkování:**

Děkuji vedoucímu bakalářské práce Ing. Lukáši Březinovi, Ph.D. za vedení, odborné konzultace a cenné rady poskytnuté při zpracovávání bakalářské práce.

**Prohlášení:**

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně na základě svých vědomostí, rad a pokynů vedoucího bakalářské práce, poskytnutých materiálů a odborných konzultací.

V Brně, dne 25. 5. 2012

---

Luboš Pemeľ

## **Obsah:**

<b>1. Úvod</b> .....	9
<b>2. Kinematika</b> .....	9
<b>Přímá kinematika</b> .....	11
<b>Inverzní kinematika</b> .....	13
Vektorová metoda řešení inverzní kinematiky.....	14
Numerické metody řešení inverzní kinematiky.....	14
Numerické řešení soustavy transcendentních rovnic.....	14
Aproximační metody.....	15
Optimalizační metody.....	15
<b>3. Dynamika</b> .....	16
<b>Newton-Eulerova metoda</b> .....	16
<b>Euler-Lagrangeova metoda</b> .....	16
<b>Hamiltonova metoda</b> .....	16
<b>Metoda virtuálních prací</b> .....	16
<b>4. Řízení</b> .....	17
<b>Stabilita</b> .....	17
<b>Linearizace zpětnou vazbou</b> .....	18
<b>Robustní řízení</b> .....	18
<b>Adaptivní řízení</b> .....	19
<b>Metoda gain – scheduling</b> .....	19
<b>5. Pohony</b> .....	19
<b>Motory</b> .....	19
Elektrické.....	19
Hydraulické.....	21
Pneumatické.....	21
<b>Převody</b> .....	21
Převod ozubenými koly.....	21
Převod kuličkovým šroubem.....	21
Řemenové a řetězové převody.....	22
<b>6. Kinematika rovinného manipulátoru</b> .....	22
<b>Počet stupňů volnosti</b> .....	23
<b>Vytvoření simulačního modelu</b> .....	23
<b>Kinematika manipulátoru</b> .....	23
Dopředná kinematika.....	23
Inverzní kinematika.....	24
Modelování v Simulinku.....	27
Simulace.....	29
<b>7. Model manipulátoru</b> .....	29
Model v MSC Adams.....	29
Co-simulace.....	33
<b>Model manipulátoru s motory</b> .....	37
Model v Adams.....	37
Model v Simulinku.....	37
<b>8. Závěr</b> .....	41
<b>Seznam použité literatury</b> .....	42
<b>Seznam použitých symbolů</b> .....	44
<b>Seznam příloh</b> .....	45

<b>Příloha 1.....</b>	<b>46</b>
<b>Příloha 2.....</b>	<b>47</b>



## 1. Úvod:

Průmyslové roboty a manipulátory jsou obecně stroje vykonávající původně lidskou práci. Průmyslový robot je v ISO 8373 definován jako „*automaticky řízený, opětovně programovatelný, víceúčelový manipulátor pro činnost ve třech nebo více osách, který může být buď upevněn na místě, nebo mobilní k užití v průmyslových automatických aplikacích*“ [5]. Ostatní zařízení tohoto charakteru, včetně těch s nulovou inteligencí (např. s pevně nastavenými pracovními cykly) tedy spadají pod manipulátory. Definice však existuje více a pojmy se často zaměňují.

Toto téma je velmi aktuální, nasazování průmyslových robotů a manipulátorů do výroby (automatizace výroby) zvyšuje její produktivitu, ekonomičnost a kvalitu.

Samotná práce se potom skládá ze dvou částí – rešeršní a praktické. V rešeršní části práce jsou popsány základní kroky návrhu průmyslových robotů a manipulátorů. V praktické části je potom tento návrh s využitím simulačních nástrojů MSC Adams a Simulink demonstrován na příkladu jednoduchého rovinného manipulátoru. Využití práce je plánováno především pro výukové účely.

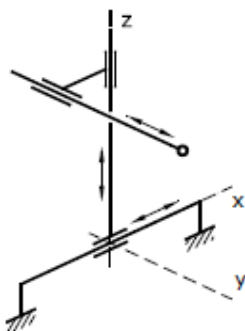
## 2. Kinematika:

Tématem této kapitoly se podrobněji zabývá [1]. Kinematiku manipulátorů popisujeme pomocí kinematických řetězců, vytvořených posloupností kinematických dvojic, sestavených z jednotlivých členů a vazeb mezi nimi. Kinematické dvojice bývají u běžných manipulátorů obvykle pouze posuvné a rotační.

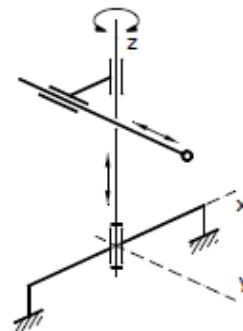
Podle kinematického řetězce lze manipulátory rozdělit na:

- 1.) sériové – mají otevřený kinematický řetězec (každý člen řetězce je připojen pouze jednou kinematickou dvojicí)
- 2.) paralelní – mají uzavřený kinematický řetězec (každý člen řetězce je připojen nejméně dvěma kinematickými dvojicemi)
- 3.) hybridní – kombinují oba předchozí typy kinematických řetězců

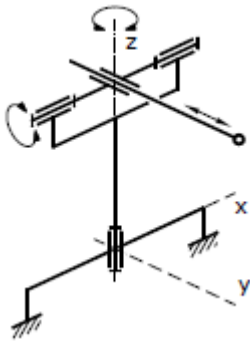
Typ kinematického řetězce při návrhu manipulátoru volíme zejména podle pracovního prostoru, výrobních nákladů, požadované trajektorie koncového efektoru a přesnosti manipulátoru. Nejrozšířenější typy kinematických řetězců jsou (obr. 2.2 - 2.8):



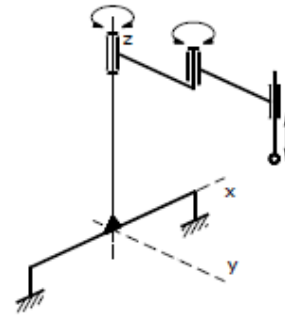
Obr. 2.1 – stojanový kartézský  
cylindrický (převzato z [1])



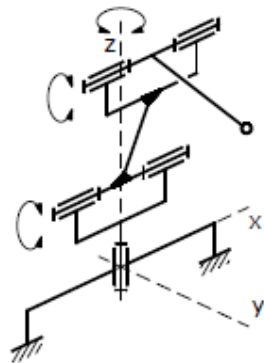
Obr. 2.2 – stojanový  
(převzato z [1])



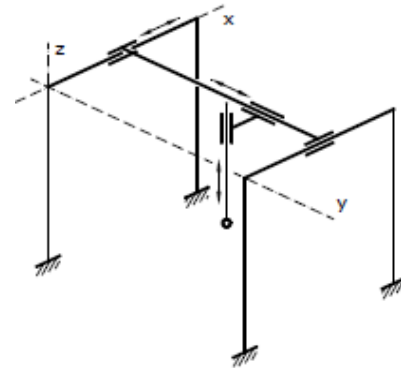
Obr. 2.3 – angulární  
(převzato z [1])



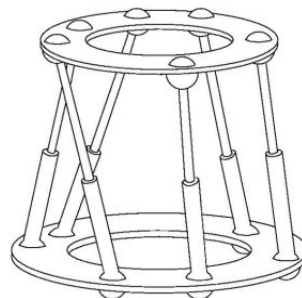
Obr. 2.4 – SCARA  
(převzato z [1])



Obr. 2.5 – PUMA  
(převzato z [1])

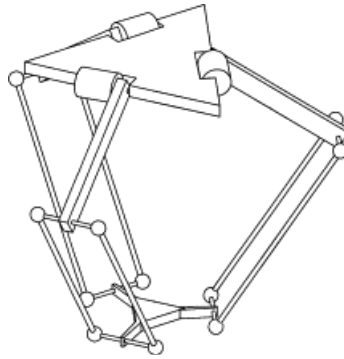


Obr. 2.6 – gantry (portálový)  
(převzato z [1])



Obr. 2.7 – Stewartova plošina,  
(převzato z

<http://www.iri.upc.edu/research/webprojects/cuikweb/Stewartgough/stewartgough.php> )



Obr. 2.8 - Delta robot  
(převzato z [6])

Koncový efektor je pracovní částí robotu. Podle požadované úlohy může být uchopovací (např. chapadla), technologický (např. hlavice pro řezání) a kontrolní (např. kamerové systémy).

Každá kinematická dvojice má přiřazenou zobecněnou souřadnici (tzv. kloubovou proměnnou)  $q_i$ , která udává úhel natočení nebo vzájemné posunutí navazujících členů. Všechny zobecněné souřadnice mechanismu s  $n$  stupni volnosti tvoří vektor zobecněných souřadnic:

$$\mathbf{q} = [q_1, q_2, \dots, q_n]^T \quad (1)$$

„Poloha a orientace nástroje je pak dána spojeným vektorem polohy  $\mathbf{p}$  a orientace  $\mathbf{o}$  nástroje – tzv. komplexním (rozšířeným) vektorem polohy  $\mathbf{w}$  (Tool Configuration Vector), který má šest souřadnic a je definován:“ [1]

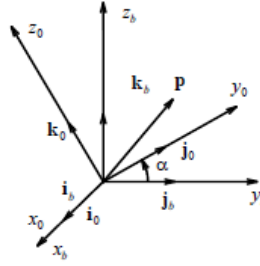
$$\mathbf{w} = [\mathbf{p}, \mathbf{o}]^T = [p_x, p_y, p_z, o_x, o_y, o_z]^T \quad (2)$$

Kromě těchto klasických ortogonálních souřadnic se používají tzv. homogenní souřadnice, kde se doplňuje čtvrtá souřadnice. Ta má pro bod hodnotu 0 a pro vektor hodnotu 1.

Známe-li vektor zobecněných souřadnic a hledáme komplexní vektor polohy, řešíme přímou úlohu kinematiky. Naopak pokud známe komplexní vektor polohy a hledáme vektor zobecněných souřadnic, řešíme inverzní úlohu kinematiky.

### **Přímá kinematika:**

Určíme orientaci a polohu efektoru ze známých zobecněných proměnných. Problém řešíme transformací souřadnic z lokálních souřadných systémů do globálního souřadného systému. Lokální souřadný systém je přiřazen každému členu manipulátoru a vyjadřuje se v něm zobecněná proměnná daného členu. Orientaci a polohu efektoru však chceme znát v globálním souřadném systému vztahenému k pracovnímu prostoru.



Obr. 2.9 – Rotace souřadných systémů  
(převzato z [1])

Pro odvození transformačních matic uvažujme dva souřadné systémy:  $x_0, y_0, z_0$  a  $x_b, y_b, z_b$ , oba s jednotkovými vektory  $\mathbf{i}, \mathbf{j}, \mathbf{k}$  ve směrech příslušných os, a vektor  $\mathbf{p} = (p_{x0}, p_{y0}, p_{z0})$  určený v souřadném systému  $x_0, y_0, z_0$  (obr. 9). Chceme-li vektor  $\mathbf{p}$  vyjádřit v souřadném systému  $x_b, y_b, z_b$ , dostaneme tuto soustavu rovnic:

$$\begin{aligned} p_{xb} &= \mathbf{p} \cdot \mathbf{i}_b \\ p_{yb} &= \mathbf{p} \cdot \mathbf{j}_b \\ p_{zb} &= \mathbf{p} \cdot \mathbf{k}_b \end{aligned} \quad (3)$$

Vektor  $\mathbf{p}$  přepíšeme do tvaru  $\mathbf{p} = \mathbf{i}_0 \cdot p_{x0} + \mathbf{j}_0 \cdot p_{y0} + \mathbf{k}_0 \cdot p_{z0}$  a soustavu rovnic vyjádříme maticově:

$$\begin{bmatrix} p_{xb} \\ p_{yb} \\ p_{zb} \end{bmatrix} = \begin{bmatrix} \mathbf{i}_b \cdot \mathbf{i}_0 & \mathbf{i}_b \cdot \mathbf{j}_0 & \mathbf{i}_b \cdot \mathbf{k}_0 \\ \mathbf{j}_b \cdot \mathbf{i}_0 & \mathbf{j}_b \cdot \mathbf{j}_0 & \mathbf{j}_b \cdot \mathbf{k}_0 \\ \mathbf{k}_b \cdot \mathbf{i}_0 & \mathbf{k}_b \cdot \mathbf{j}_0 & \mathbf{k}_b \cdot \mathbf{k}_0 \end{bmatrix} \begin{bmatrix} p_{x0} \\ p_{y0} \\ p_{z0} \end{bmatrix} \quad (4)$$

$$\begin{bmatrix} \mathbf{i}_b \cdot \mathbf{i}_0 & \mathbf{i}_b \cdot \mathbf{j}_0 & \mathbf{i}_b \cdot \mathbf{k}_0 \\ \mathbf{j}_b \cdot \mathbf{i}_0 & \mathbf{j}_b \cdot \mathbf{j}_0 & \mathbf{j}_b \cdot \mathbf{k}_0 \\ \mathbf{k}_b \cdot \mathbf{i}_0 & \mathbf{k}_b \cdot \mathbf{j}_0 & \mathbf{k}_b \cdot \mathbf{k}_0 \end{bmatrix} = \mathbf{R} \quad (5)$$

Pro natočení kolem osy  $x$  o úhel  $\alpha$  (obr. 2.1) pak s využitím vztahu

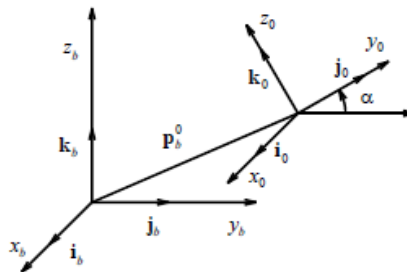
$$\mathbf{u} \cdot \mathbf{v} = |\mathbf{u}| \cdot |\mathbf{v}| \cdot \cos \alpha \quad (6)$$

kde  $\mathbf{u}, \mathbf{v}$  jsou různoběžné vektory a úhel, který svírají, dostaneme transformační matici

$$\mathbf{R}_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{bmatrix} \quad (7)$$

Transformační matice pro natočení kolem os  $y$  a  $z$  se odvodí analogicky.

Nyní uvažujme kromě vyřešeného pootočení kolem osy  $x$  i vzájemné posunutí souřadných systémů určené vektorem  $\mathbf{p}$ .



Obr. 2.10 – Posunutí a rotace souřadných systémů  
(převzato z [1])

Pro transformaci souřadnic bodu  $\mathbf{P} = [x_0, y_0, z_0]$  zadaných v souřadném systému  $x_0, y_0, z_0$  do souřadného systému  $x_b, y_b, z_b$  platí následující rovnice:

$$\begin{bmatrix} x_b \\ y_b \\ z_b \end{bmatrix} = \mathbf{R}_x \cdot \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} + \begin{bmatrix} p_{xb} \\ p_{yb} \\ p_{zb} \end{bmatrix} \quad (8)$$

Použijeme-li homogenní souřadnice, můžeme rovnici (8) přepsat do tvaru:

$$\begin{bmatrix} x_b \\ y_b \\ z_b \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_x & \begin{bmatrix} p_{xb} \\ p_{yb} \\ p_{zb} \end{bmatrix} \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ y_0 \\ z_0 \\ 1 \end{bmatrix} \quad (9)$$

$$\mathbf{A} = \begin{bmatrix} \mathbf{R}_x & \begin{bmatrix} p_{xb} \\ p_{yb} \\ p_{zb} \end{bmatrix} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (10)$$

Matici  $\mathbf{A}$  nazýváme homogenní transformační maticí. Je zřejmé, že matici  $\mathbf{A}$  je možno odvodit pro jakékoliv natočení a pro jakýkoliv posuv.

„Při dodržení konvence rozmístování souřadných systémů (Denavit a Hartenberg, 1995), je možno sestavovat vzájemné transformační matice automaticky.“ [1]

Provedeme pouze dosazení Denavit-Hartenbergových parametrů  $\vartheta_i$ ,  $d_i$ ,  $a_i$ ,  $\alpha_i$  do univerzální transformační matice: [1]

$$\mathbf{A}_{i-1,i} = \begin{bmatrix} \cos \vartheta_i & -\sin \vartheta_i \cdot \cos \alpha_i & \sin \vartheta_i \cdot \sin \alpha_i & a_i \cdot \cos \vartheta_i \\ \sin \vartheta_i & \cos \vartheta_i \cdot \cos \alpha_i & -\cos \vartheta_i \cdot \sin \alpha_i & a_i \cdot \sin \vartheta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (11)$$

$\vartheta_i$  ... úhel mezi osami  $x_{i-1}$  a  $x_i$  při otáčení kolem  $z_{i-1}$

$d_i$  ... nejkratší vzdálenost (normála) mezi osami  $x_{i-1}$  a  $x_i$ , kladný směr ve směru  $z_{i-1}$

$a_i$  ... nejkratší vzdálenost (normála) mezi osami  $z_{i-1}$  a  $z_i$ , kladný směr ve směru  $x_i$

$\alpha_i$  ... úhel mezi osami  $z_{i-1}$  a  $z_i$  při otáčení kolem  $x_i$

Pokud pro transformaci souřadnic z lokálních souřadných systémů do globálního souřadného systému nestačí uvažovat jen jeden posuv, nebo jen jednu rotaci, matice se násobí v pořadí odpovídajícímu pořadí uvažovaných pohybů. Takto vytvoříme celkovou transformační matici  $\mathbf{T}_{bn}$ :

$$\mathbf{T}_{bn}(q_1, q_2, \dots, q_n) = \mathbf{A}_{b0} \cdot \mathbf{A}_{01}(q_1) \cdot \mathbf{A}_{12}(q_2) \dots \cdot \mathbf{A}_{n-1,n}(q_n) \quad (12)$$

Řešení přímé úlohy kinematiky je dáno vztahem:

$$\begin{bmatrix} x_b \\ y_b \\ z_b \\ 1 \end{bmatrix} = \mathbf{T}_{bn}(q_1, q_2, \dots, q_n) \cdot \begin{bmatrix} x_n \\ y_n \\ z_n \\ 1 \end{bmatrix} \quad (13)$$

Přepočítání  $\mathbf{T}_{bn}$  na  $\mathbf{w}$  řeší např. [7].

### Inverzní kinematika:

Úlohou inverzní kinematiky je ze známého komplexního vektoru polohy  $\mathbf{w}$  určit vektor zobecněných souřadnic  $\mathbf{q}$ . Tato úloha je poměrně komplikovaná, musíme v reálném čase řešit nelineární transcendentní soustavu rovnic, která může mít i nekonečně mnoho řešení. Potýkáme se také s problémem singulárních pozicí.

„V singulárních pozicích efektor ztrácí jeden či více stupňů volnosti, kinematické rovnice se stávají lineárně závislé nebo se určité řešení stává nedefinovaným.“ [2]

Podle [1] se metody řešení inverzní kinematiky dají rozdělit na:

- 1.) Vektorové
- 2.) Numerické:
  - a.) numerické řešení soustavy transcendentních rovnic
  - b.) aproximační
  - c.) optimalizační

### **Vektorová metoda řešení inverzní kinematiky:**

Podle [1] je výhodou vektorové (geometrické) metody je především rychlost výpočtu a snadná programovatelnost, výsledkem jsou totiž explicitně vyjádřené jednotlivé  $q_i$ . Používá se proto hlavně pro řízení polohy v reálném čase. Hlavní slabinou je nutnost odvodit řešení pro každou konkrétní kinematickou strukturu. Kinematickou dvojici, jejíž kloubovou proměnnou řešíme, promítneme do vhodné roviny tak, abychom mohli najít řešení použitím trigonometrických vztahů mezi články. Tato metoda bude demonstrována na příkladě rovinného manipulátoru v kapitole 6.

### **Numerické metody řešení inverzní kinematiky:**

Tyto metody mají výhodu ve své univerzálnosti, ale výpočetně jsou náročnější než vektorová metoda. Podle [1] se numerické metody používají zejména u velkých CAD systémů.

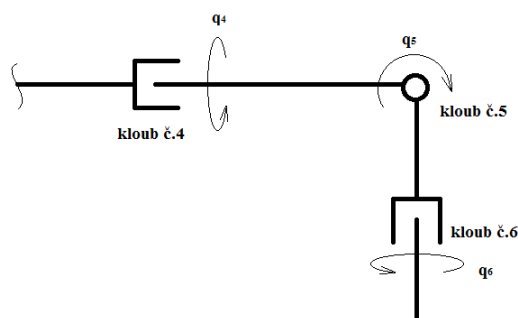
### **Numerické řešení soustavy transcendentních rovnic:**

Jednou z metod vedoucích k numerickému řešení transcendentních rovnic je metoda inverzní transformace. Předpokládejme, že máme zadaný komplexní vektor polohy  $w$ . Pak máme vyčíslenou i celkovou transformační matici  $T_{bn}$ . Sestavíme-li jednotlivé transformační matice  $A_{i-1,i}$  jako funkce jednotlivých zobecněných souřadnic  $q_i$  (je výhodné použít Denavit-Hartenbergovu konvenci) a matici  $T_{bn}$  vyjádříme symbolicky jako funkci všech zobecněných souřadnic manipulátoru  $q_1$  až  $q_n$ , můžeme na základě rovnosti

$$T_{bn} = T_{bn}(q_1, q_2, \dots, q_n) \quad (14)$$

sestavit soustavu rovnic pro určení vektoru zobecněných souřadnic  $q$ .

Další metoda numerického řešení transcendentních rovnic inverzní kinematiky spočívá v rozdělení úlohy inverzní kinematiky na inverzní úlohu polohy a inverzní úlohu orientace. Tato metoda je odvozena pro manipulátor s šesti klouby, z nichž poslední tři tvoří Stanfordské rameno.



Obr. 2.11 – Stanfordské rameno

Nejprve vyřešíme inverzní úlohu polohy. Jejím řešením jsou první tři zobecněné souřadnice  $q_1, q_2$  a  $q_3$ . Pak se dostaneme k další části řešení, k inverzní úloze orientace, odkud vypočítáme  $q_4, q_5$  a  $q_6$ .

#### Aproximační metody:

*„Řešení soustavy kinematických rovnic numerickými metodami je relativně obtížné. Určité zjednodušení představují numerické metody, které aproximují průběh polohy a orientace koncového bodu v okolí výchozího známého bodu – aproximační metody“ [8]*

Základní aproximační metodou je Newton-Raphsonova metoda. Tato metoda vychází z Taylorova rozvoje 1.řádu a využívá jakobiánu  $\mathbf{J}$ . Ten umožňuje ze změny jednotlivých kloubových proměnných  $\Delta q_i$  přibližně vypočítat změnu komplexního vektoru polohy  $\Delta \mathbf{w}$ . Jakobián, vystupující v druhém členu Taylorova rozvoje, je maticí parciálních derivací jednotlivých transformačních funkcí. Jako neznámé ve výpočtu vystupují změny jednotlivých kloubových proměnných oproti výchozí poloze  $\Delta q_i$  a z nich pak vypočteme další vektor zobecněných souřadnic  $\mathbf{q}$ , který určuje další výchozí polohu. Výpočet se opakuje, dokud výsledná chyba polohování není menší než požadovaná přesnost polohování. Při řešení inverzní kinematiky ale potřebujeme z vektoru  $\mathbf{w}$  vypočítat  $\mathbf{q}$ , potřebujeme tedy inverzi jakobiánu  $\mathbf{J}^{-1}$ . Tu však obecně nelze určit ( $\mathbf{J}$  nemusí tvořit čtvercovou matici) a pro výpočet se používá pseudoinverze jakobiánu  $\mathbf{J}^+$ .

Další možností je použít výpočet podle Newton-Raphsonovy metody upravený tak, že se namísto pseudoinverze jakobiánu  $\mathbf{J}^+$  používá transpozice jakobiánu  $\mathbf{J}^T$ . Tato úprava vychází z principu virtuálních prací. Newton-Raphsonova metoda se dá použít i pro určení rychlostí a zrychlení v kinematických vazbách, jakobián pak transformuje rychlost (zrychlení) efektoru na rychlosti (zrychlení) ve vazbách.

*„Iterační metoda často vyžaduje mnoho výpočtů a navíc nezaručuje konvergenci ke správnému řešení. Je slabá zejména pokud je manipulátor v blízkosti singulárních a degenerovaných konfigurací.“ [2]*

#### Optimalizační metody:

Optimalizační metody jsou obecně založeny na hledání extrému funkce. Pro nás bude touto funkcí chyba polohování  $E = f(\mathbf{q})$ , která je součtem chyby polohy  $\Delta p$  a chyby orientace  $\Delta o$ :

$$E = \Delta p + \Delta o \quad (15)$$

Optimálním řešením je v tomto případě takové  $\mathbf{q}$ , pro které je hodnota  $E$  minimální.

Jednou z nejpoužívanějších optimalizačních metod je metoda CCD (cyclic coordinate descent). Je to heuristická iterační metoda, která od efektoru k bázi postupně prochází jednotlivé kinematické vazby a fiktivně nastavuje  $q_i$  tak, aby bylo dosaženo minimální chyby v poloze a orientaci. Nejprve se tedy nastaví poloha kin. vazby nejbližší efektoru, ta se poté zablokuje a nastavuje se směrem od efektoru druhá kin. vazba. Hledáme vždy jen jednu neznámou  $q_i$ , což lze řešit analyticky. Výpočet se opakuje, dokud výsledná chyba polohování není menší než požadovaná přesnost polohování.

### **3. Dynamika:**

*„Zatímco kinematické rovnice popisují pohyb robotu bez uvážení sil a momentů způsobujících pohyb, dynamické rovnice popisují vztah mezi silami a pohybem.“ [3]*

#### **Newton-Eulerova metoda:**

Tato metoda je podrobněji popsána např. v [1]. Pracuje s rovnicemi silové a momentové rovnováhy. Při sestavování rovnic uvažujeme tíhovou sílu, setrvačnou sílu, reakční sílu (zobecněnou sílu) a akční sílu. Postupujeme od posledního článku. Akční silou je síla, kterou působí na  $i$  - tý článek článek číslo  $i+1$  (blíže k efektoru) nebo u posledního článku síla, kterou působí efektor na poslední článek (např. tíha břemene, síla způsobená nástrojem). Naší snahou je však určit zobecněné síly na jednotlivých článcích. *„Zobecněná síla je zjednodušeně průmět reakční síly do směru pohybu nebo momentu do osy otáčení a v případě mechanismů je to zjednodušeně ta složka reakce v pohybové jednotce, kterou musí vyvinout její pohon, aby se daný článek mechanismu pohyboval po zadané trajektorii, s předepsanou rychlostí a zrychlením. Ostatní složky reakce v pohybové jednotce jsou zachyceny vlastní pohybovou jednotkou, tj. vedením, ložisky apod.“ [1]* Metoda převádí řešení problému dynamiky celého manipulátoru na řešení dynamiky jednotlivých článků. Výhodou metody je zejména to, že dává úplné informace o kinematických veličinách a silových poměrech na všech článcích.

#### **Euler-Lagrangeova metoda:**

Je důkladně rozebrána např. v [2]. Spočívá v sestavení Lagrangeových pohybových rovnic II. druhu pro jednotlivé kin. vazby. V těchto rovnicích vystupují derivace kinetické a potenciální energie (resp. jejich rozdíl – Lagrangiánu) podle času a podle jednotlivých zobecněných souřadnic a zobecněné síly ve vazbách, které zahrnují potřebnou sílu či moment pohonu v dané vazbě. Výhodou metody oproti Newton-Eulerově metodě je hlavně snazší implementace řízení.

#### **Hamiltonova metoda:**

Metoda je rozebrána v [9]. Metoda popisuje dynamiku systému Hamiltonovými pohybovými rovnicemi. Vystupuje v nich Hamiltonián, neboli součet kinetické a potenciální energie systému, který je funkcí zobecněných souřadnic a zobecněných hybností. Tato metoda je blízká Euler–Lagrangeově metodě, místo jedné diferenciální rovnice druhého řádu pro každou vazbu však získáme dvě diferenciální rovnice prvního řádu pro každou vazbu.

#### **Metoda virtuálních prací:**

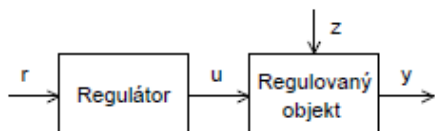
Podle principu virtuálních prací je práce všech vnějších a setrvačných sil či momentů při virtuálních posunutích rovna nule. Tento princip zachycuje obecná rovnice dynamiky, kterou musíme sestavit pro jednotlivé kin. vazby (poháněné). Metoda je podle [13] výhodná při řešení dynamiky paralelních manipulátorů, protože do výpočtu nezahrnuje reakční síly a momenty v nepoháněných kin. vazbách, které paralelní manipulátory vždy obsahují.



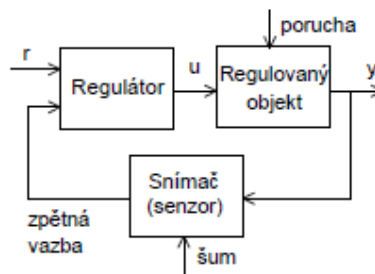
## 4. Řízení:

„Teorie řízení je soubor poznatků z aplikované matematiky, umožňující analýzu a syntézu, tj. návrh takových vhodných regulátorů, aby zpětnovazební soustava byla stabilní a měla požadované dynamické chování, např. velikost překmitu, koeficient tlumení, ustálenou chybu apod.“ [4] Problematika řízení je velmi obsáhlá. Tato práce se bude tématem zabývat jen okrajově.

Aby se daná veličina  $y$  (poloha, rychlost, ...) při provozu manipulátoru dostala vhodným způsobem na žádanou hodnotu  $r$ , je potřeba zavést zpětnovazební řízení, které nastavuje hodnotu řízené veličiny podle její skutečné hodnoty na hodnotu žádanou. „Zpětnovazební řízení kompenzuje neurčitost parametrů soustavy a do určité míry i jejich nelinearity, dále kompenzuje vliv poruchových veličin, např. změny zatěžovacího momentu u elektrických pohonů, vliv změny teploty okolí u otopných soustav, kolísání síťového napětí apod.“ [4] Kromě zpětnovazebního řízení existuje i řízení v otevřené smyčce. To nastavuje hodnotu řízené veličiny bez informací o její skutečné hodnotě, výše zmíněné vlivy proto způsobují, že se řízená veličina nemusí ustálit na žádané hodnotě.



Obr. 4.1 – Řízení v otevřené smyčce (převzato z [4])



Obr. 4.2 – Řízení v uzavřené smyčce (převzato z [4])

Manipulátory jsou nelineární dynamické soustavy. Metodika návrhu vhodných regulátorů je propracována hlavně pro lineární systémy a pro nelineární systémy není žádná obecná metoda návrhu vhodného regulátoru. Existuje však mnoho metod, díky kterým jsme schopni navrhnout regulátor vhodný pro dané úkoly a typy manipulátorů.

### Stabilita:

Stabilita je nutnou podmínkou pro správnou funkci lineárních i nelineárních regulačních obvodů. Manipulátory jsou nelineární dynamické soustavy, budeme se ale zabývat i stabilitou lineárních systémů, se kterými některé metody pro návrh řízení manipulátorů pracují.

Lineární systém je stabilní, pokud se po ukončení působení vstupního signálu vrátí zpět do počátečního stavu. „U nestabilních soustav výstupní signál buď trvale monotónně narůstá (u reálných soustav na maximálně dosažitelnou hodnotu z hlediska fyzikální realizovatelnosti), nebo kmitá s rostoucí amplitudou (u reálných soustav opět na maximálně dosažitelnou hodnotu). Stabilní soustava má odezvu na jednotkový skok buď aperiodickou, jsou-li všechny kořeny charakteristické rovnice reálné záporné, nebo odezvu kmitavou, je-li aspoň jedna dvojice kořenů komplexních, se zápornou reálnou částí. Podmínkou stability zpětnovazební soustavy je, aby charakteristická rovnice měla všechny kořeny záporné, pokud jsou reálné, nebo komplexní se zápornou reálnou částí.“ [4] Pro posouzení stability lineárních systémů se kromě výpočtu kořenů charakteristické rovnice používá například Routh–Horwitzovo nebo Nyquistovo kritérium (popsáno v [4]).

U nelineárních systémů je situace složitější, existuje více definic stability. Pro nelineární soustavy neplatí princip superpozice, existuje více možných rovnovážných stavů atd. Tyto vlastnosti způsobují, že stabilita nelineárního systému nezávisí jen na vnitřní struktuře systému, ale i na jeho počátečním stavu a vstupním signálu. Problematikou stability nelineárních systémů se hlouběji zabývá [11].

Uvedeme základní definice stability nelineárních systémů. Jednou z nich je Ljapunova lokální stabilita. Tato definice zjednodušeně říká, že daný rovnovážný stav soustavy je stabilní, jestliže po malém vychýlení z tohoto stavu soustava zůstane v jeho blízkosti (nemusí se tedy vrátit do tohoto stavu). Další definicí je asymptotická stabilita. Podle ní je daný rovnovážný stav systému stabilní, je-li Ljapunovsky lokálně stabilní a vrací-li se systém do rovnovážného stavu, ze kterého byl vychýlen. Dále se rozlišuje asymptotická stabilita globální a v uzavřené oblasti.

Pro vyšetřování stability používáme přímou Ljapunovou metodou nebo Popovo kritérium. Ljapunovou metodou posuzujeme definice stability rovnovážného bodu pomocí Ljapunovy funkce. Popovo kritérium stability pro nelineární systémy rozděluje soustavu na lineární a nelineární část. Systém je globálně asymptoticky stabilní, pokud kořeny charakteristické rovnice jeho lineární části leží v levé části Gaussovy roviny a pokud neliarity systému vyhovují metodou určené podmínce.

### **Linearizace regulované soustavy:**

Rovnice popisující soustavu se nazývají charakteristické rovnice. Charakteristickými rovnicemi lineární soustavy jsou lineárními diferenciální rovnice. „V lineární diferenciální rovnici se hledaná funkce  $y$  vyskytuje pouze lineárně a nikdy se nevyskytují součiny hledané funkce s jejími derivacemi, ani součiny derivací této funkce.“ [10] Dá se zapsat v tomto tvaru:

$$y^{(n)} + a_{n-1}(x) \cdot y^{(n-1)} + \dots + a_1(x) \cdot y' + a_0(x) \cdot y = f(x) \quad (16)$$

Hlavní myšlenkou této metody je linearizovat pohybové rovnice manipulátoru v okolí pracovního bodu. Pro návrh stabilní regulace v okolí tohoto bodu pak můžeme použít jednu z metod pro návrh regulátoru lineární soustavy. Pracovních bodů zpravidla volíme více a regulaci zajišťuje regulátor navržený pro oblast, ve které se manipulátor právě pohybuje. Oblasti, ve kterých jsou jednotlivé regulátory použitelné, však mohou být těžko zjistitelné nebo malé.

### **Linearizace zpětnou vazbou:**

Metodou linearizace zpětnou vazbou se zabývá [2]. Metoda spočívá ve vytvoření regulátoru pro chybu, která je popsána lineární diferenciální rovnicí. [2] Moment je tedy dán zákonem řízení, jež lze rozdělit na dvě části: první část určuje potřebný moment ze zákona řízení pro otevřenou smyčku, další pak moment upravuje tak, aby se redukovala chyba v polohování a rychlosti. Metoda však nemusí být dostatečně robustní na zohlednění nejistoty parametrů soustavy.

### **Robustní řízení:**

Návrh regulátoru metodou robustního řízení je založen na modelu obsahujícím nejistoty. Regulátor je tedy navržen tak, aby byla soustava stabilní pro parametry s nejistotou v jistém rozsahu. „Nejistotu můžeme zohlednit v mnoha parametrech, třeba v hmotnosti objektu uchopeného koncovým efektozem. Navrhujeme pak například řízení, které je efektivní pro rozsah hmotnosti 1 - 10 kg.“ [2]

### Adaptivní řízení:

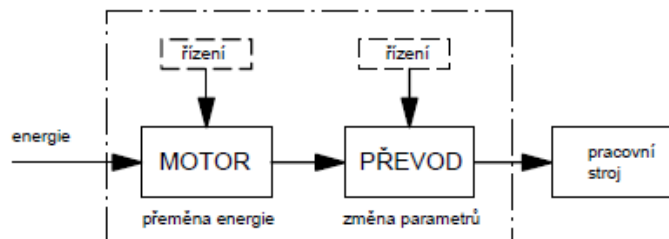
Narozdíl od klasických zpětnovazebních regulátorů s pevně nastavenými parametry adaptivní regulátory mění způsob zpracování regulační odchylky. Adaptivní zpětná vazba mění parametry regulátoru podle kvality regulace. Metoda adaptivního řízení se používá zejména pro řízení robotů s nejistotami v parametrech měnících se v závislosti na čase.

### Metoda gain – scheduling:

Tato metoda spočívá ve vytvoření časově proměnné lineární aproximace dynamiky robota v okolí zvolených pracovních bodů. Body volíme tak, aby pokryly pracovní prostor robota. Na tento přibližný popis dynamického chování soustavy aplikujeme vhodnou metodu návrhu regulátoru používanou pro lineární systémy. Parametry regulátoru jsou získávány interpolací parametrů určených pro jednotlivé pracovní body.

## 5. Pohony:

*„Pohon je zařízení pro přeměnu energie, včetně řízení této přeměny (parametrů  $M$ ,  $\omega$ , resp.  $F$ ,  $v$ ), určené k uvedení poháněného stroje do požadovaného stavu (předepsaným způsobem). Pohony lze klasifikovat podle druhu energie, která je využita k přeměně na energii mechanickou v pracovním stroji na: spalovací, elektrické, hydraulické, pneumatické.“ [1]*



Obr. 5.1 – Schéma pohonu  
(převzato z [1])

Na pohony manipulátorů jsou kladeny velké nároky zejména v přesnosti nastavování polohy. Zpravidla se proto používají pohony umožňující řízení polohy, tzv. servopohony (někdy servomotory).

### Motory:

U motorů v pohonech manipulátorů jsou důležité hlavně tyto parametry: velikost, váha, výkon a rozsah rychlostí.

### Elektrické:

Jsou nejpoužívanější skupinou motorů pro pohony manipulátorů. Hlavní výhody jsou v dostupném zdroji energie, snadné údržbě, vysoké účinnosti, dobré říditelnosti a v čistotě provozu. Nejsou ale vhodné pro práci v prostředí, kde hrozí požár.

Nejpopulárnějšími jsou díky své dobré říditelnosti stejnosměrné motory, tedy DC motor a EC motor. Připomeneme v náznaku principy činnosti těchto motorů, abychom z nich mohli vyvodit jejich výhody a nevýhody.

DC motor má ve statoru permanentní magnety a na rotoru vinutí, jež je napájeno přes komutátor. Magnetické pole vinutí a statorových magnetů je orientováno

souhlasně, tím motor generuje moment a rotor se otočí. Komutátor svou konstrukcí zajišťuje napájení těch cívek v rotorovém vinutí, které jsou „pod“ magnety na statoru. Na vinutí rotoru tak vznikne opět magnetické pole orientované souhlasně s magnetickým polem statoru (v případě dvoupólového motoru se změní polarita napájení rotorového vinutí).

EC (elektronicky komutovaný) motor funguje podobně jako DC motor. Je napájen stejnosměrným proudem, točivé magnetické pole ve vinutí však narozdíl od DC motoru nevzniká díky komutátoru. Jeho funkci přebírá tranzistorový měnič, který točivé magnetické pole vytváří tak, že podle polohy rotoru postupně přivádí napětí na jednotlivé cívky vinutí ve statoru. Motor proto musí mít snímače polohy rotoru (např. Hallovy sondy). Permanentní magnety umístěné na rotoru se pak otáčejí společně s magnetickým polem vygenerovaným ve vinutí statoru.

Nevýdou DC motoru je právě komutátor, který způsobuje ztráty elektrické (na přechodovém odporu mezi kartáči a lamelami) i ztráty mechanické (třením). Další předností EC motoru oproti DC motoru zajišťuje umístění vinutí na statoru, kde se dá lépe zajistit odvod tepla. Vinutí na rotoru navíc zvyšuje setrvačný moment motoru. Jednodušší DC motory jsou však levnější.

Poměrně časté jsou krokové motory. Mezi základní typy krokových motorů patří VR motory (reluktanční), PM motory (s permanentními magnety) a hybridní motory. Způsob jejich činnosti je podobný. Mají nastavené sekvence, ve kterých se postupně přivádí impulsy budícího napětí na jednotlivé póly ve statoru motoru. Rotor se pak natáčí buď díky interakci magnetických polí statorového vinutí a permanentních magnetů na rotoru (PM motory), nebo zaujímá polohu s nejmenší energií, tj. s nejmenším magnetickým odporem - reluktancí (VR motory), či využívá obou principů (hybridní motory). Poloha se tedy řídí počtem impulsů, rychlost řídíme jejich frekvencí. Odtud plyne, že k řízení motoru nepotřebujeme žádné snímače polohy. Nevýhodou je, že při malých rychlostech je pohyb rotoru nespojitý, což působí rázy ve stroji. Musíme se také vyhnout frekvenci impulsů, která je rovna rezonanční frekvenci stroje. Naopak při vyšších rychlostech motor ztrácí moment. Při zapojení příliš velké zátěže může dojít ke ztrátě kroků, která způsobí nepřesnosti v řízení. Krokové motory se používají pro svou dobrou říditelnost hlavně v pohonech malých manipulátorů, běžně i bez převodů.

Lineární motory umožňují přímočarý pohyb bez použití převodů. V pohonech manipulátorů se z elektrických lineárních motorů používají zejména krokové a synchronní motory. Princip jejich činnosti odpovídá klasickým rotačním verzím těchto motorů, stator i rotor jsou však rozvinuty do roviny. Tyto motory se začaly ve větší míře používat až v posledních letech. Hlavní výhodou je fakt, že lineární motory mohou pracovat bez převodů. Kromě zjednodušení konstrukce manipulátorů to přináší vyšší dosažitelné rychlosti (nedochází např. k rozkmitání šroubu u převodu kuličkovým šroubem) a také lepší přesnost (převody mají vůle a opotřebovávají se). Lineární motory oproti rotačním motorům s převodem kratší přechodové jevy a rychleji se dostanou do ustálené polohy s minimálním překmitem. Rotační motory s převodem mají naopak větší tuhost a zpravidla také sílu zdvihu.

Méně používané jsou v pohonech manipulátorů střídavé motory. Mají oproti stejnosměrným větší výkon při stejných rozměrech, vyžadují ale složitější řízení a výkonovou elektroniku. Hlavními představiteli této skupiny jsou asynchronní a synchronní motory, principem jejich činnosti se zabývá [12].

### Hydraulické:

Pro některé aplikace jsou hydraulické motory vhodnější pro svůj velký výkon. Zpravidla na výstupu poskytují veliké síly (momenty) při nižších rychlostech, používají se proto bez převodů. Mohou pracovat i v prostředí, kde hrozí požár. Jsou ale větší než el. motory a ke své činnosti vyžadují další zařízení (např. čerpadla). Navíc vyžadují údržbu, mají nižší účinnost, vlastnosti pracovní kapaliny závisí na teplotě pracovního prostředí a prosakující pracovní kapalina znečišťuje okolí.

Princip jejich činnosti je jednoduchý – práci koná tekutina (většinou minerální olej) stlačená ve válci, jehož objem volíme podle tlaku přiváděné kapaliny a požadovaného výkonu. Hydraulické motory mohou být lineární, rotační i kyvné. Lineární hydraulické motory jsou tvořeny jedním válcem, rotační a kyvné jich mají více.

### Pneumatické:

Tento druh motorů není příliš používaný, využívá se hlavně k vykonávání jednoduchých pohybů (např. rozevření a sevření čelistí na koncovém efektoru). Dosahují poměrně dobrých rychlostí, pro svou jednoduchost jsou nenáročné na údržbu, mohou pracovat i v prostředí s rizikem požáru či výbuchu a nepoškodí se při přetížení. Mají však zásadní nedostatek: je velmi obtížné řídit polohu v celém rozsahu pohybu a obtížné je i řízení rychlosti. To je zapříčiněno poddajností stlačeného vzduchu.

### Převody:

Nejpoužívanějšími motory jsou elektromotory, které poskytují velké rychlosti pohybu a malý moment (sílu). Pro práci robotů je však potřeba nižších rychlostí, ale velkých momentů (sil). Pomocí převodů také můžeme měnit druh pohybu, např. z rotačního pohybu na translační (tomto případě mluvíme o neidentické transformaci, pokud se druh pohybu nemění jedná se o identickou transformaci). Abychom zajistili žádané parametry pohonu, musíme do pohonu zařadit převod. V manipulátorech se často objevují zejména:

#### Převod ozubenými koly:

Umožňuje nám měnit rychlost a moment rotačního pohybu, polohu a orientaci os rotace. Často používané jsou planetové převody, které jsou oproti klasickým převodovkám účinnější a menší.

#### Převod kuličkovým šroubem:

Převádí rotační pohyb na pohyb posuvný. „*Prakticky zcela nahradily pohybové šrouby se smykovým třením. Tyto šrouby v důsledku nízké účinnosti byly samosvorné. Valivé šrouby mají velmi nízký součinitel valivého tření a nejsou tudíž samosvorné, je nutné aplikovat brzdu.*“ [1]

Obdobou převodu kuličkovým šroubem je v poslední době upřednostňovaný převod válečkovým šroubem (obr. 5.2). Je přesnější, přenesení větší síly a momenty, pracuje s většími rychlostmi a zrychleními, ale je dražší.



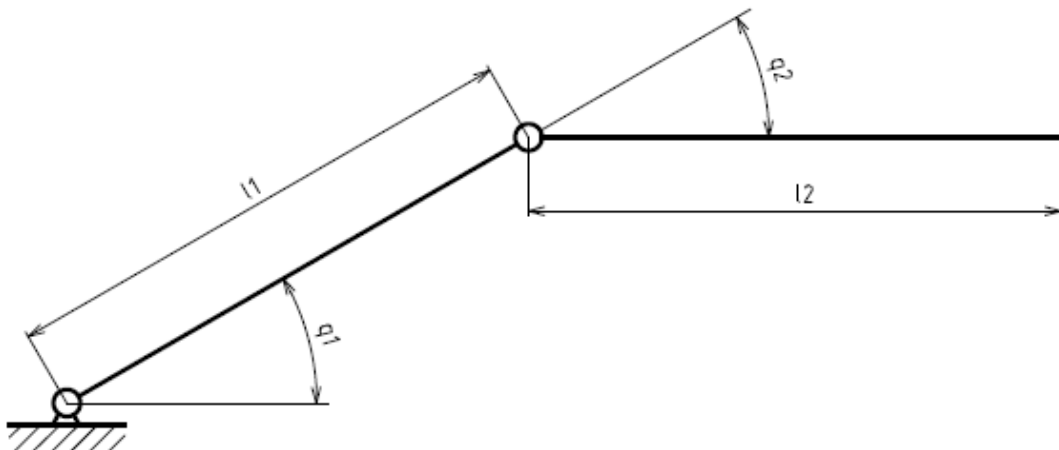
Obr. 5.2 – Planetový válečkový šroub  
(převzato z <http://www.directindustry.com/prod/power-jacks/planetary-roller-screws-22145-55999.html> )

Řemenové a řetězové převody:

Používají se pro přenos kroutícího momentu mimo osu rotace motoru. Řemenové převody se používají pro přenos menších momentů při větších rychlostech, řetězové naopak pro přenos velkých momentů při malých rychlostech.

## **6. Kinematika rovinného manipulátoru:**

Základní kroky návrhu manipulátoru ukážeme na příkladu rovinného manipulátoru se dvěma rotačními vazbami (obr. 6.1). Jeho kinematický řetězec se v praxi objevuje spíše jako součást složitějších řetězců.



Obr. 6.1 – Rovinný manipulátor

Připojením dalšího členu posuvnou kinematickou dvojjící dostaneme kinematický řetězec manipulátoru typu SCARA (Obr. 2.4). Ten se obvykle používá pro

manipulaci s menšími objekty, zejména pro tzv. pick and place (vezmi a polož) aplikace v montáži či paletizaci. Vyniká vysokou rychlostí pohybu a přesností.

Dalším kin. řetězcem odvozeným z rovinného manipulátoru se dvěma rotačními vazbami je řetězec manipulátoru typu PUMA (Obr. 2.5). PUMA je nejpoužívanějším typem manipulátoru pro svařování a montáž.

### **Počet stupňů volnosti:**

Mechanismus se skládá celkem ze tří těles, spojených dvěma rotačními vazbami. První rotační vazba spojuje základní těleso (bázi) s článkem, který je spojen druhou rotační vazbou s dalším článkem. Každá rotační vazba přitom odebrává dva stupně volnosti. Mechanismus uvažujeme pouze v rovině. Můžeme tedy napsat, že:

$$i_v = 3$$

$$n_t = 3$$

$$n_v = 2$$

$$\xi_1, \xi_2 = 2$$

kde

$i_v$  je počet stupňů volnosti v uvažovaném prostoru,  $n_t$  je počet těles (včetně základního),

$n_v$  je počet vazeb a  $\xi_i$  je počet stupňů volnosti odebraný  $i$  - tou vazbou.

Počet stupňů volnosti  $i$  je dán vztahem:

$$i = i_v \cdot (n_t - 1) - \sum_{i=1}^{n_v} \xi_i \quad (17)$$

$$i = 3 \cdot (3 - 1) - 4 = 2 \text{ } ^\circ V$$

Po dosazení vyšlo, že řešený manipulátor má dva stupně volnosti.

### **Vytvoření simulačního modelu:**

Pro analýzu dynamiky a modelování manipulátoru využijeme na škole dostupný MSC Adams. Tento software se používá hlavně v automobilovém průmyslu a umožňuje analýzu kinematiky a dynamiky komplexních systémů.

Model vytvořený v Adams budeme exportovat do Simulinku, který je rovněž velice rozšířený. Umožňuje přehledně modelovat rozmanité dynamické systémy blokovými schématy. Použijeme jej pro výpočet inverzní kinematiky, zadávání trajektorií a modelování pohonů.

Oba softwary jsou tzv. multi-domain, dávají nám možnost řešit rozmanité problémy (např. mechaniku, řízení a elektroniku) na jednom modelu, čímž jsou vhodné pro mechatronický přístup návrhu.

### **Kinematika manipulátoru:**

#### **Dopředná kinematika:**

Protože používáme Denavit-Hartenbergův způsob notace, provedeme pouze dosazení parametrů do univerzálních transformačních matic:

Do transformační matice první kin. vazby  $\mathbf{A}_{b,1}$  za  $\vartheta_1$  (úhel mezi osami  $x_{i-1}$  a  $x_i$  při otáčení kolem  $z_{i-1}$ ) dosadíme  $q_1$  a za  $a_i$  (nejkratší vzdálenost (normála) mezi osami  $z_{i-1}$  a  $z_i$ , kladný směr ve směru  $x_i$ ) dosadíme  $l_1$ :

$$A_{b,1} = \begin{bmatrix} \cos(q_1) & -\sin(q_1) & 0 & l_1 \cdot \cos(q_1) \\ \sin(q_1) & \cos(q_1) & 0 & l_1 \cdot \sin(q_1) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (18)$$

Do transformační matice první kin. vazby  $A_{1,2}$  za  $\vartheta_1$  dosadíme  $q_2$  a za  $a_i$  dosadíme  $l_2$ :

$$A_{1,2} = \begin{bmatrix} \cos(q_2) & -\sin(q_2) & 0 & l_2 \cdot \cos(q_2) \\ \sin(q_2) & \cos(q_2) & 0 & l_2 \cdot \sin(q_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (19)$$

Výsledná transformační matice je jejich součinem:

$$T_{b,2} = A_{b,1} \cdot A_{1,2} \quad (20)$$

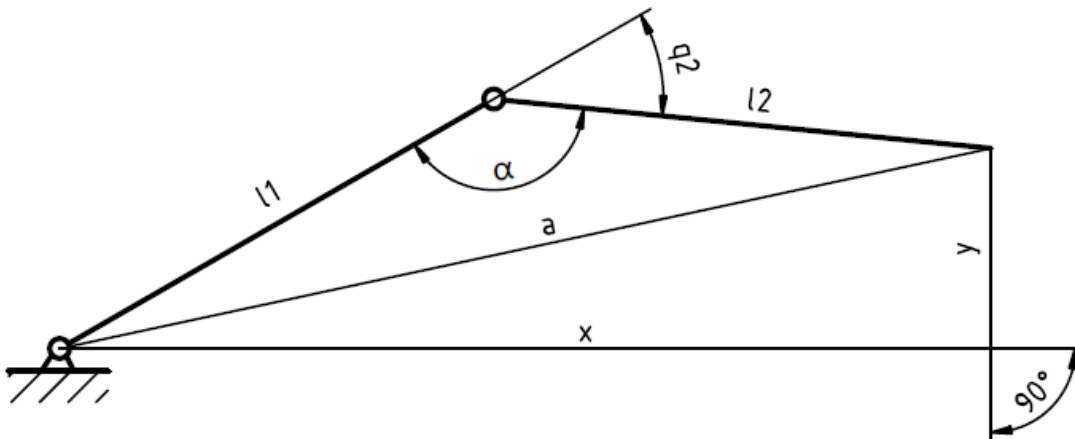
Poloha v globálním souřadném systému je dána vztahem:

$$\begin{bmatrix} x_b \\ y_b \\ z_b \\ 1 \end{bmatrix} = T_{bn}(q_1, q_2, \dots, q_n) \cdot \begin{bmatrix} x_n \\ y_n \\ z_n \\ 1 \end{bmatrix} \quad (21)$$

Protože  $x_n, y_n, z_n$  jsou nulové, poloha v globálních souřadnicích  $[x_b; y_b]$  je vyčíslena v celkové transformační matici prvky  $[(1,4);(2,4)]$ . Toho jsme využili ve funkcích pro výpočet  $x_b$  a  $y_b$  v Matlabu (příloha 1).

#### Inverzní kinematika:

Pro sestavení rovnic inverzní kinematiky použijeme vektorovou metodu. Začneme zobecněnou souřadnicí  $q_2$ :



Obr. 6.2 – Určení  $q_2$

Rozměr  $a$  určíme z Pythagorovy věty:

$$a = \sqrt{x^2 + y^2} \quad (22)$$

Pro trojúhelník na Obr. 6.2 napíšeme kosinovou větu:

$$a^2 = l_1^2 + l_2^2 - 2 \cdot l_1 \cdot l_2 \cdot \cos \alpha \quad (23)$$

Do rovnice (22) dosadíme rovnici (23) a vyjádříme úhleh  $\alpha$ :

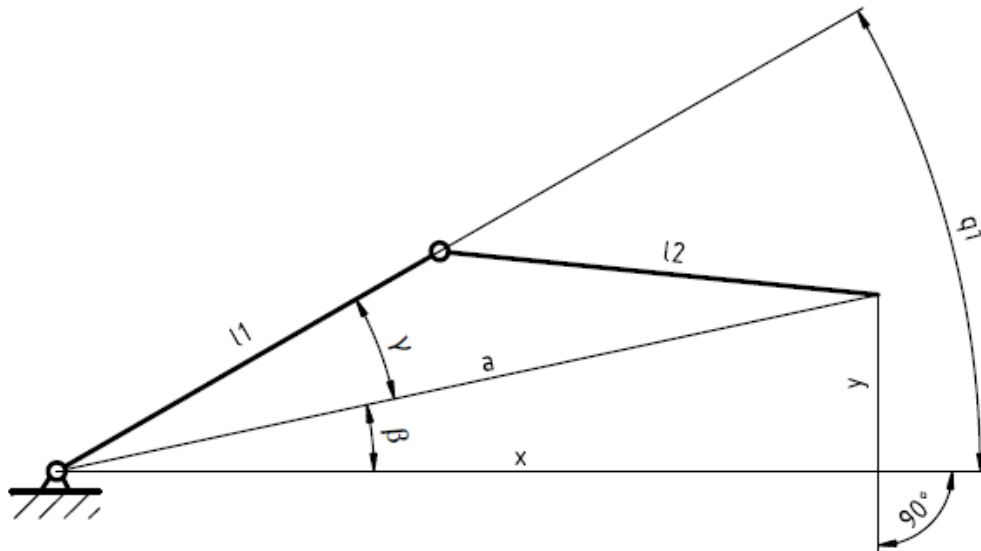


$$\alpha = \arccos \frac{l_2^2 + l_1^2 - (x^2 + y^2)}{2 \cdot l_1 \cdot l_2} \quad (24)$$

Z obr. 6.2 vidíme, že hodnotu  $q_2$  získáme takto:

$$q_2 = \pi - \alpha = \pi - \arccos \frac{l_2^2 + l_1^2 - (x^2 + y^2)}{2 \cdot l_1 \cdot l_2} \quad (25)$$

Při řešení zobecněné souřadnice  $q_1$  budeme postupovat dle obr. 6.3:



Obr. 6.3 – Určení  $q_1$

Pro trojúhelník se stranami  $a$ ,  $x$ ,  $y$  platí:

$$\operatorname{tg} \beta = \frac{y}{x} \quad (26)$$

Odtud vyjádříme úhel  $\beta$ :

$$\beta = \operatorname{arctg} \frac{y}{x} \quad (27)$$

Pokračujeme stanovením úhlu  $\gamma$ . Pro trojúhelník se stranami  $l_1$ ,  $l_2$  a napíšeme kosinovou větu:

$$l_2^2 = a^2 + l_1^2 - 2 \cdot a \cdot l_1 \cdot \cos \gamma \quad (28)$$

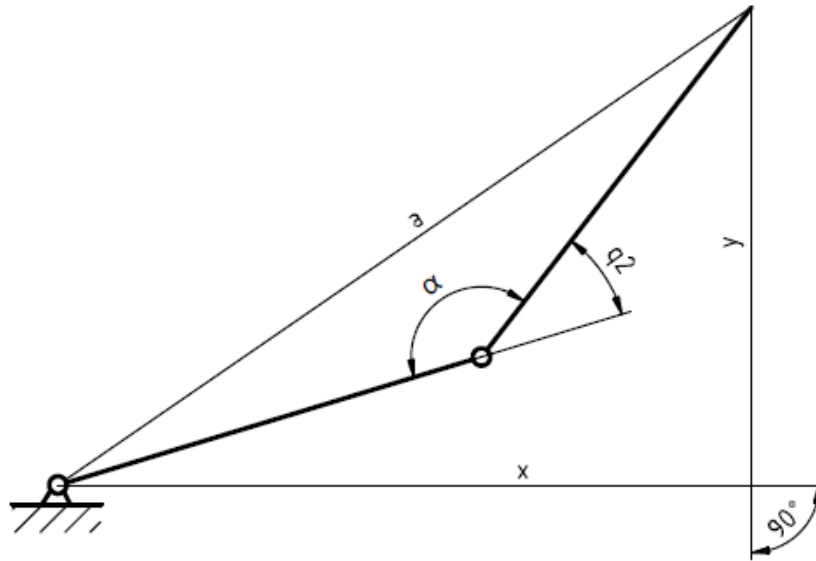
Dosadíme do ní rovnici (22) a vyjádříme úhel  $\gamma$ :

$$\gamma = \arccos \frac{l_1^2 - l_2^2 + (x^2 + y^2)}{2 \cdot l_1 \cdot \sqrt{(x^2 + y^2)}} \quad (29)$$

Úhel  $q_1$  je součtem úhlů  $\beta$  a  $\gamma$ :

$$q_1 = \beta + \gamma = \operatorname{arctg} \frac{y}{x} + \arccos \frac{l_1^2 - l_2^2 + (x^2 + y^2)}{2 \cdot l_1 \cdot \sqrt{(x^2 + y^2)}} \quad (30)$$

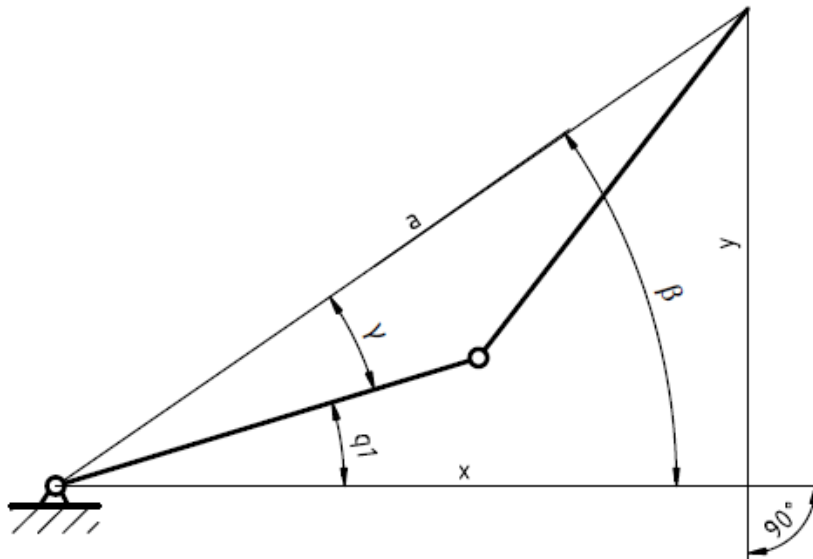
Rovinný manipulátor má pro každou pozici mimo hranice pracovního prostoru a mimo bodu  $[0,0]$  dvě řešení inverzní kinematiky. Doposud jsme vyřešili pouze tzv. elbow up konfiguraci. Nyní určíme hodnoty zobecněných souřadnic pro elbow down konfiguraci. Úhel  $q_2$  stanovíme z obr. 6.4:



Obr. 6.4 – Určení  $q_2$  v elbow down konfiguraci

Je patrné, že vztah pro určení pomocného úhlu  $\alpha$  se oproti elbow up konfiguraci nemění. Hodnotu  $q_2$  tedy vypočteme ze vztahu:

$$q_2 = \pi + \alpha = \arccos \frac{l_2^2 + l_1^2 - (x^2 + y^2)}{2 \cdot l_1 \cdot l_2} \quad (31)$$



Obr. 6.5 – Určení  $q_1$  v elbow down konfiguraci

Rovněž rovnice pro stanovení pomocných úhlů  $\beta$  a  $\gamma$  jsou v obou konfiguracích totožné. Na obr. 6.5 vidíme, že pro  $q_1$  platí:

$$q_1 = \beta - \gamma = \arctg \frac{y}{x} - \arccos \frac{l_1^2 - l_2^2 + (x^2 + y^2)}{2 \cdot l_1 \cdot \sqrt{(x^2 + y^2)}} \quad (32)$$

Teď máme všechna řešení a vytvoříme podle nich funkce pro výpočet zobecněných souřadnic v Matlabu. Použitím  $\arctg$ , tyto vztahy ale dávají špatné výsledky ve 2. a 4. kvadrantu. Tento problém jsme vyřešili odečtením  $\pi$  od původních rovnic pro  $q_1$  při splnění podmínky  $x < 0$ . Dalším problémem byl bod  $[0,0]$ , ve kterém

jsme  $q_1$  nechali stejné jako v předchozím kroku ( $q_1$  z předchozího kroku je označeno jako  $q_0$ ) a měnili jsme pouze  $q_2$ . Ze dvou možných řešení  $q_1$  a  $q_2$  vybíráme vždy to, které je bližší předchozí poloze. Funkce pro výpočet inverzní kinematiky v Matlabu jsou v příloze 2.

### Modelování v Simulinku:

Vstupem funkcí, které vytvoříme pro výpočet inverzní kinematiky budou souřadnice  $[x; y]$  jako funkce času, zadané použitím bloků z knihovny Sources. Vytvoříme:

1.) kruhovou trajektorii, popsanou rovnicemi:

$$\begin{aligned} x(t) &= 200 \cdot \sin(4 \cdot \pi \cdot t) \\ y(t) &= 200 \cdot \cos(4 \cdot \pi \cdot t) \end{aligned} \quad (33)$$

V Simulinku takovouto trajektorii vytvoříme bloky Sine Wave, kde nastavíme parametry: Amplitude na 200, Frequency na  $4 \cdot \pi$  a Phase na 0 pro  $x$  a na  $\pi/2$  pro  $y$

2.) přímou trajektorii, určenou rovnicemi:

$$\begin{aligned} x(t) &= 200 \cdot t - 100 \\ y(t) &= 200 \cdot t - 100 \end{aligned} \quad (34)$$

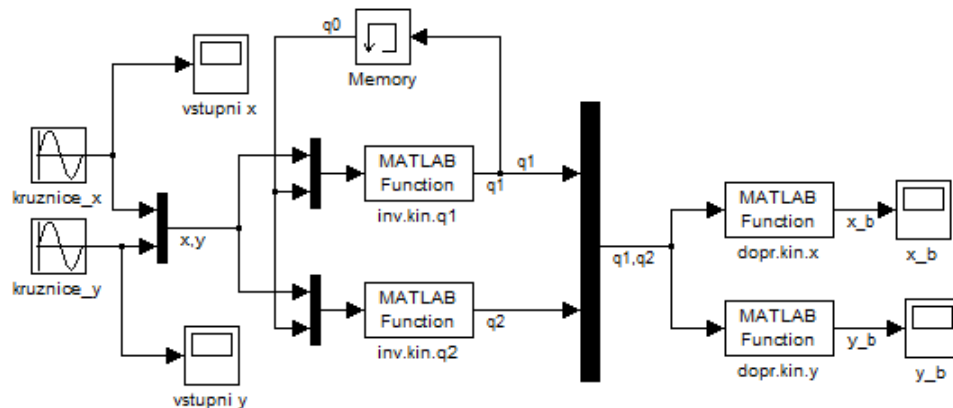
V Simulinku ji vytvoříme bloky Ramp, nastavíme parametry Slope na 200 a Initial output na -100.

3.) trajektorie kývavého pohybu:

$$\begin{aligned} x(t) &= 150 \\ y(t) &= 100 \cdot \sin(2 \cdot \pi \cdot t) \end{aligned} \quad (35)$$

V Simulinku funkci  $x$  zadáme blokem Constant s Constant value nastaveným na 150 a funkci  $y$  zadáme blokem Sine Wave, který nastavíme podle rovnice analogicky jako u kruhové trajektorie.

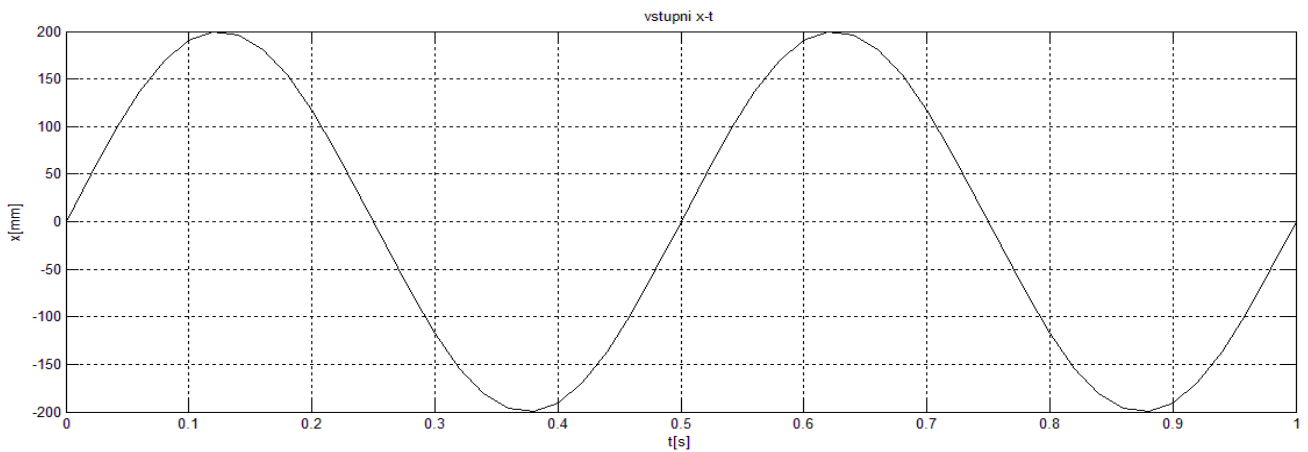
Pro výpočet dopředné a inverzní kinematiky máme vytvořeny funkce v Matlabu, do Simulinku je převedeme blokem MATLAB Function v knihovně User – Defined Functions. Nejdříve ze vstupních hodnot  $x$  a  $y$  odpovídajících poloze koncového efektoru určíme hodnoty zobecněných souřadnic  $q_1$  a  $q_2$  potřebných pro dosažení této polohy. Abychom se přesvědčili o správnosti funkcí pro výpočet inverzní kinematiky, použijeme spočtené hodnoty  $q_1$  a  $q_2$  jako vstupy a funkcemi pro výpočet dopředné kinematiky z nich určíme  $x_b$  a  $y_b$ . Průběhy vstupních hodnot  $x$  a  $y$  musí být shodné s průběhy výstupních hodnot  $x_b$  a  $y_b$ . Pro ověření správnosti funkcí inverzní a dopředné kinematiky postačí jedna trajektorie, ostatní použijeme při analýze dynamiky, vybrali jsme kruhovou trajektorii (obr. 6.6):



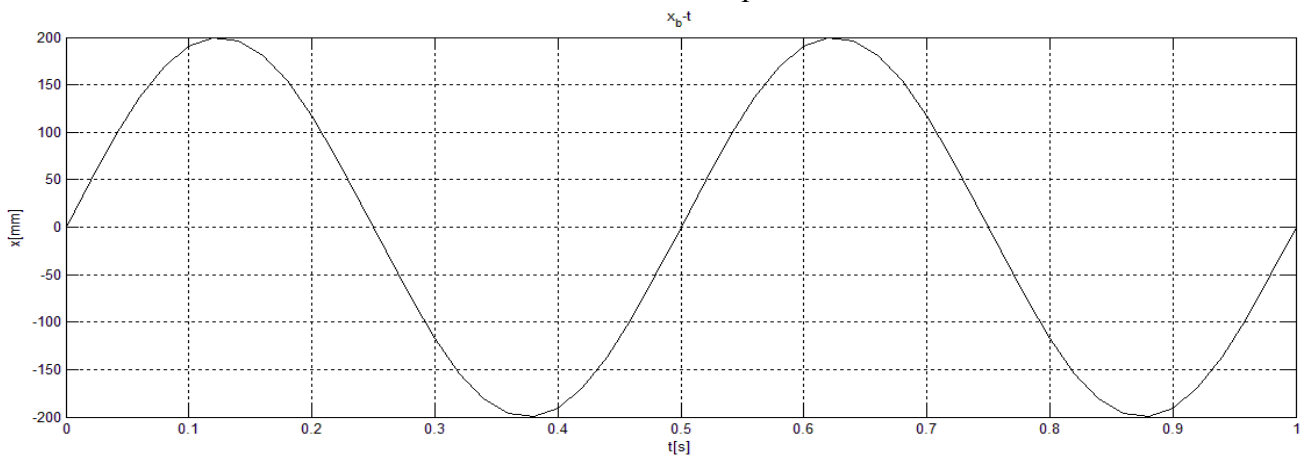
Obr. 6.6 – Ověřování výpočtů inverzní kinematiky

Simulace:

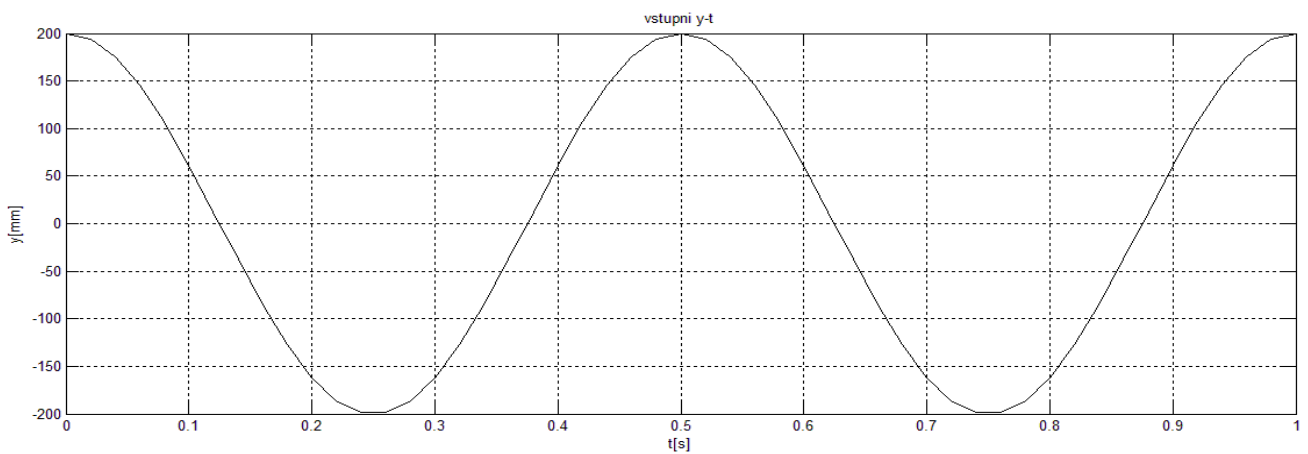
Simulační čas  $t$  jsme nastavili na 1 sekundu.



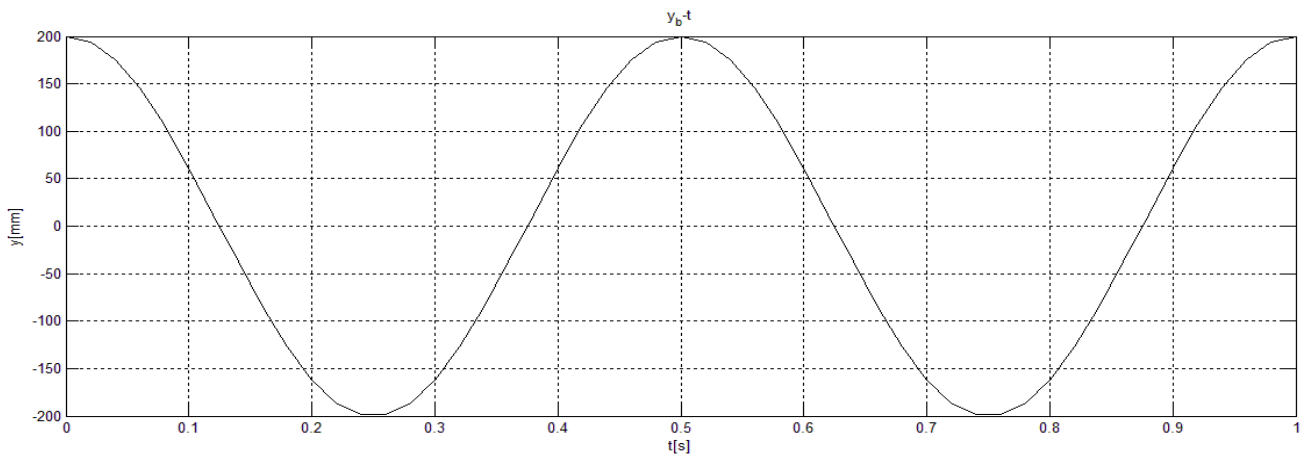
Obr. 6.6 – Závislost vstupního  $x$  na čase  $t$



Obr. 6.7 – Závislost  $x_b$  na čase  $t$



Obr. 6.8 – Závislost vstupního  $y$  na čase  $t$



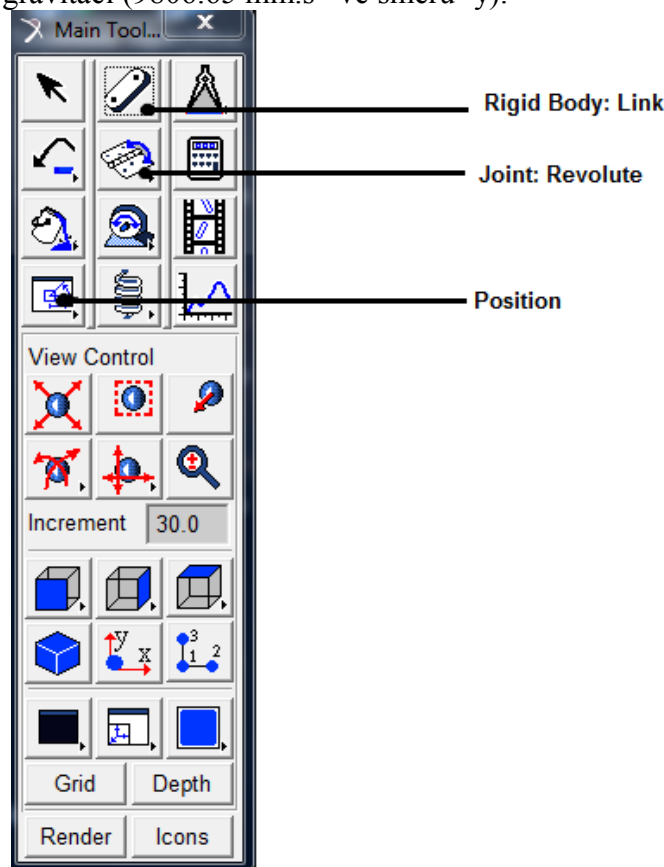
Obr. 6.9 – Závislost  $y_b$  na čase  $t$

Vstupní hodnoty  $x$  a  $y$ , generované podle rovnic popisující kruhovou trajektorii jsou shodné s hodnotami  $x_b$  a  $y_b$ , které jsou výstupem z bloků pro výpočet dopředné kinematiky (obr. 6.6 – 6.9).

## 7. Model manipulátoru:

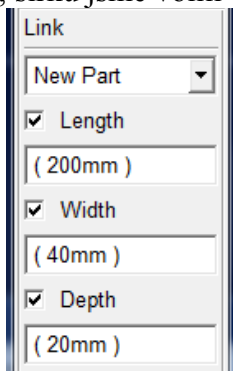
### Model v MSC Adams:

V úvodním menu zvolíme možnost vytvoření nového modelu, povolíme výchozí nastavení pro gravitaci ( $9806.65 \text{ mm.s}^{-2}$  ve směru  $-y$ ).



Obr. 7.1 – Main Toolbox MSC Adams

Články vytvoříme ikonou Rigid Body: Link (obr. 7.1). Výchozí hodnota zobecněné souřadnice  $q_1$  je  $-\pi/6$  (v Adamsu je kladný úhel definovaný ve směru otáčení hodinových ručiček), článek č.1 tedy natočíme pomocí Position/Rotate (obr. 7.1). Výchozí poloha druhého článku je vodorovná, hodnota zobecněné souřadnice  $q_2$  je tedy  $\pi/6$ . Délka obou článků je 200 mm, šířku jsme volili 40 mm a tloušťku 20 mm (obr. 7.2).

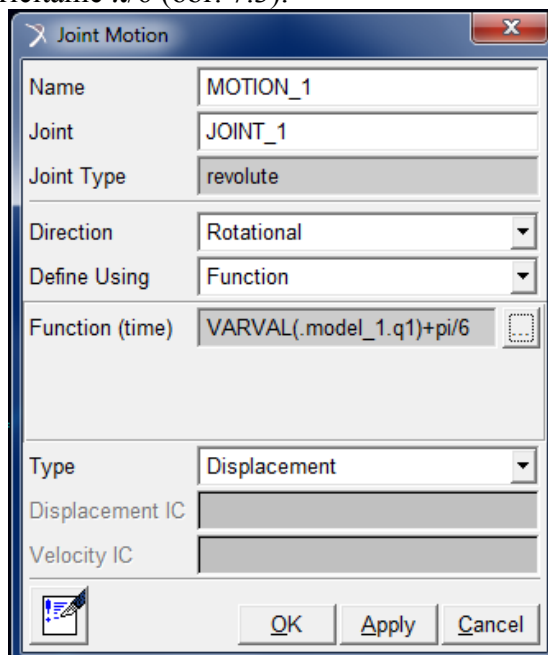


Obr. 7.2 – Rozměry článků

Následně vytvoříme rotační kinematické vazby. K tomuto účelu slouží ikona Joint: Revolute (obr. 7.1). Tyto vazby umístíme dle obr. 6.1, první vazbou vytvoříme spojení mezi zemí a prvním článkem, další spojíme oba články. Tento základní model budeme pro analýzu kinematiky a dynamiky dále rozšiřovat tak, abychom mohli ověřit správnost výpočtu inverzní kinematiky.

Nejprve definujeme stavové proměnné systému  $q_1$  a  $q_2$ , odpovídající zobecněným souřadnicím  $q_1$  a  $q_2$ , které později nadefinujeme jako vstup. Vytvoříme je příkazem New v menu Build/System Elements/State Variable a přiřadíme jim nulovou hodnotu (v poli: F(time, ...) =).

Poté do modelu přidáme předepsaný pohyb v kloubech příkazem Rotational Joint Motion, posunutí budeme zadávat prostřednictvím stavových proměnných  $q_1$  a  $q_2$  funkcí VARVAL. Musíme také zohlednit výchozí hodnotu  $q_1$ . Abychom nastavovali hodnotu  $q_1$  od nuly, přičítáme  $\pi/6$  (obr. 7.3).

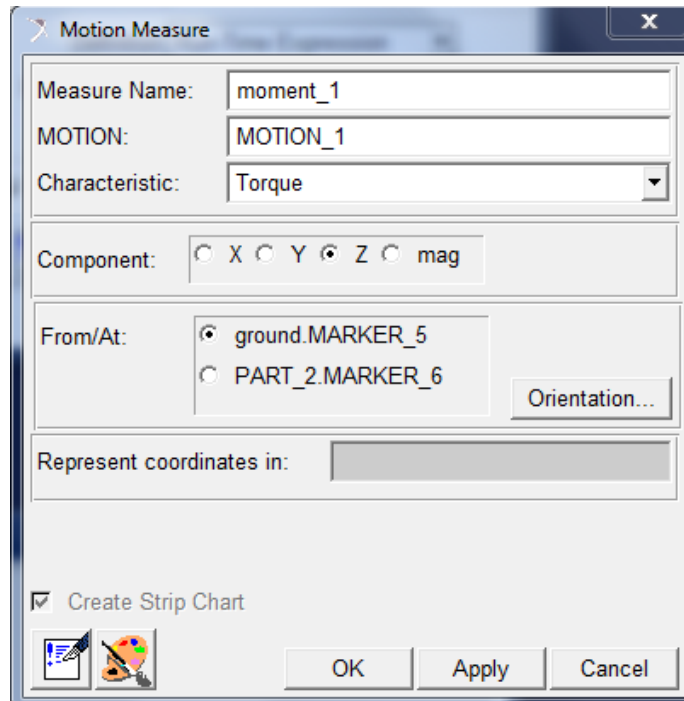


Obr. 7.3 – Zadávání pohybu

Pohyb v druhé kinematické vazbě předepíšeme analogicky.

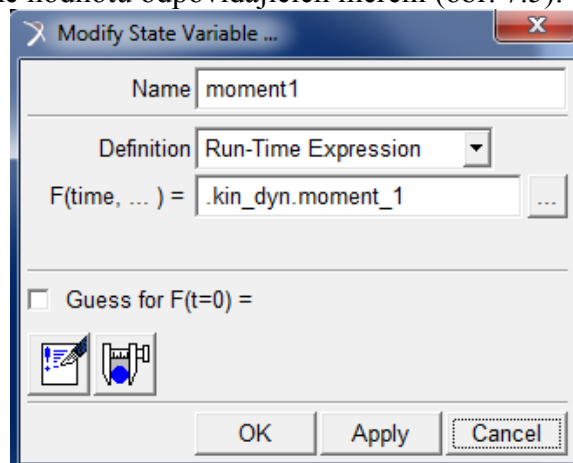
Výstupem modelu bude poloha koncového efektoru v globálních souřadnicích, zobecněné souřadnice a momenty v kin. vazbách.

Nejprve vybereme pohyb předepsaný na kin. vazbě a v menu Build/Measure/Selected Object/New vytvoříme měření příslušného momentu, např. měření momentu v první vazbě je na obr. 7.4:



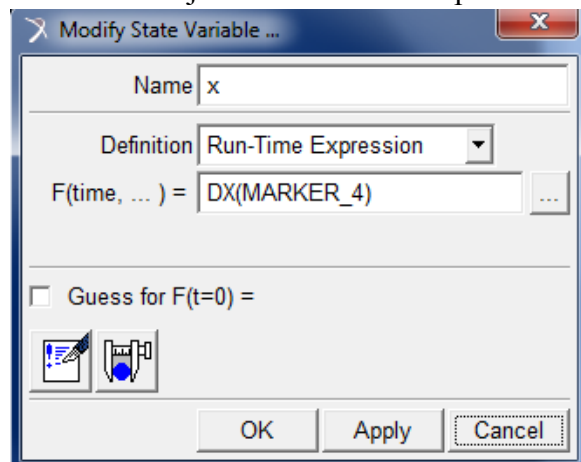
Obr. 7.4 – Měření momentu1

Následně vytvoříme stavové proměnné moment1 a moment2. Těmto stavovým proměnným přiřadíme hodnotu odpovídajících měření (obr. 7.5):



Obr. 7.5 – Stavová proměnná moment1

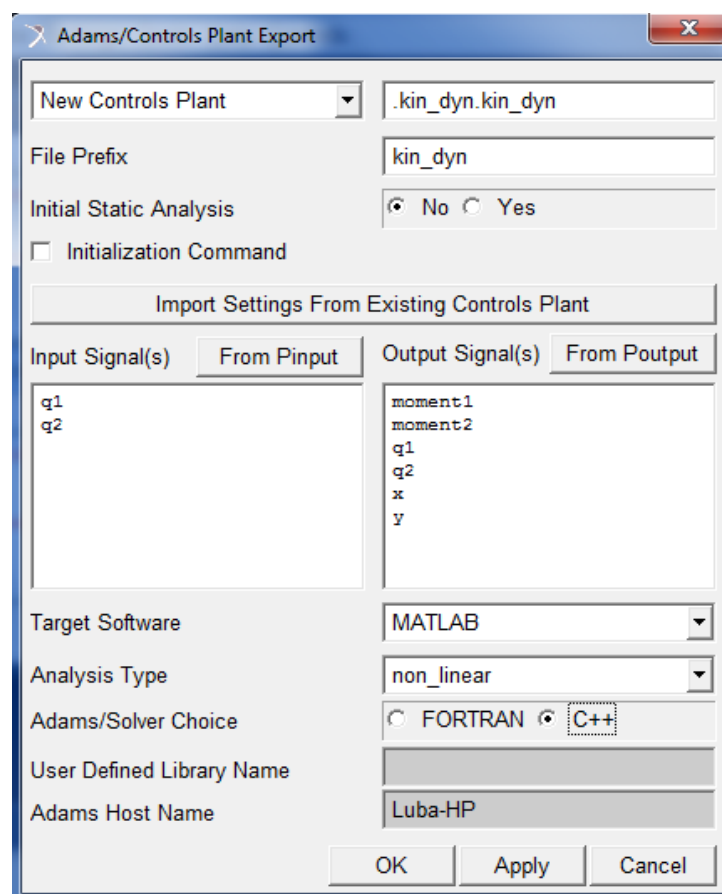
Souřadnice koncového efektoru budeme mít ve stavových proměnných  $x$  a  $y$ . K jejich vytvoření využijeme funkcí  $DX$  a  $DY$ , které vyčíslí jednotlivé souřadnice zadaného markeru v globálním souřadném systému. Na pozici koncového efektoru je  $MARKER\_4$ . Obr. 7.6 ukazuje nastavení stavové proměnné  $x$ :



Obr. 7.6 – Stavová proměnná  $x$

Měření pro stavové proměnné  $x$  a  $y$  vytvoříme kliknutím na ikonu Measures.

Posledním krokem je export modelu. Je nutné mít spuštěný modul Adams/Controls (spouští se v Tools/Plugin Manager). Samotný export provedeme s tímto nastavením dle obr. 7.7:

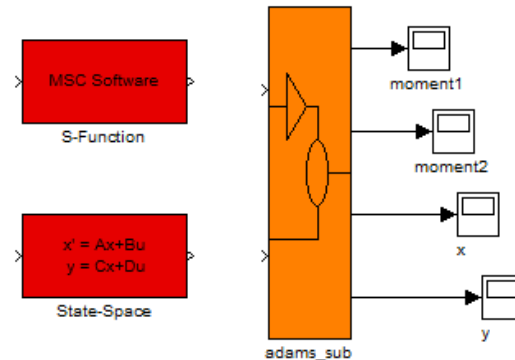


Obr. 7.7 – Export



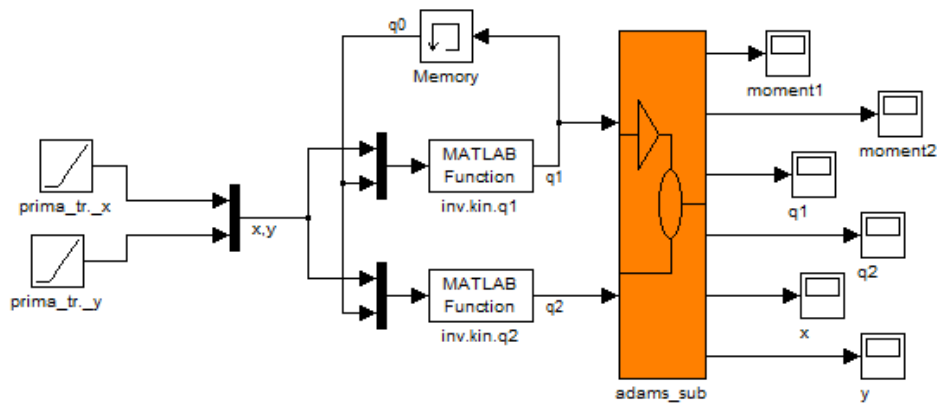
### Co-simulace:

V Matlabu otevřeme a spustíme m-file kin\_dyn.m vygenerovaný Adamsem. Na příkazový řádek napíšeme příkaz: adams\_sys, tím se vytvoří a otevrou bloky s modelem z Adamsu (obr. 7.8). Budeme používat pouze blok adams\_sub, ostatní smažeme.



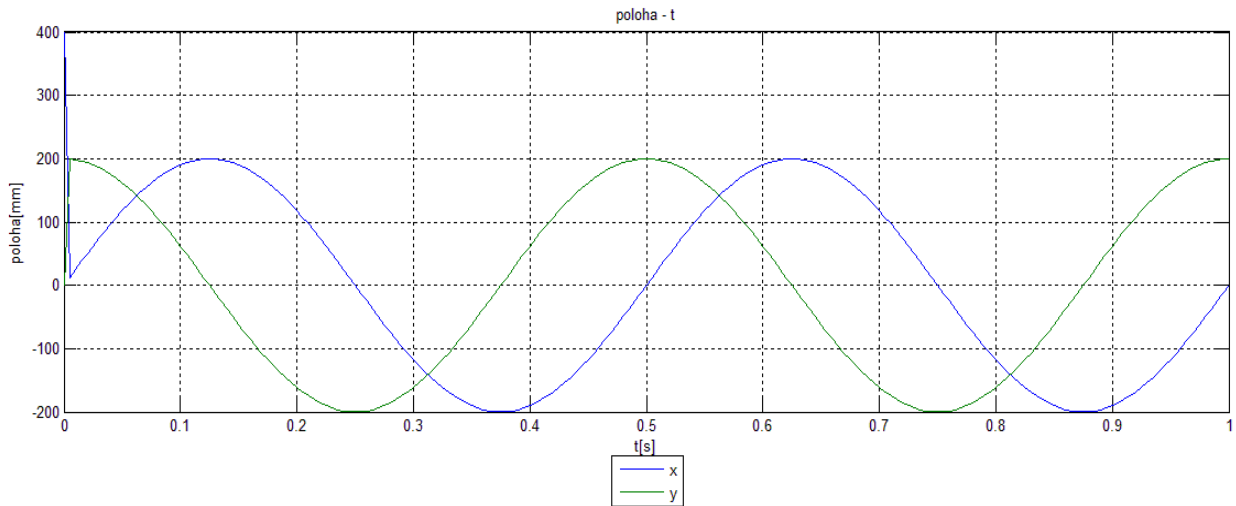
Obr. 7.8 – Bloky z Adams

Tento blok je subsystemem, otevřeme jej a bloku MSC Software nastavíme parametr Simulation mode na interactive. Blok vložíme do souboru vytvořeném pro ověření výpočtů inverzní kinematiky (obr. 6.5) a nahradíme jím funkce dopředné kinematiky.

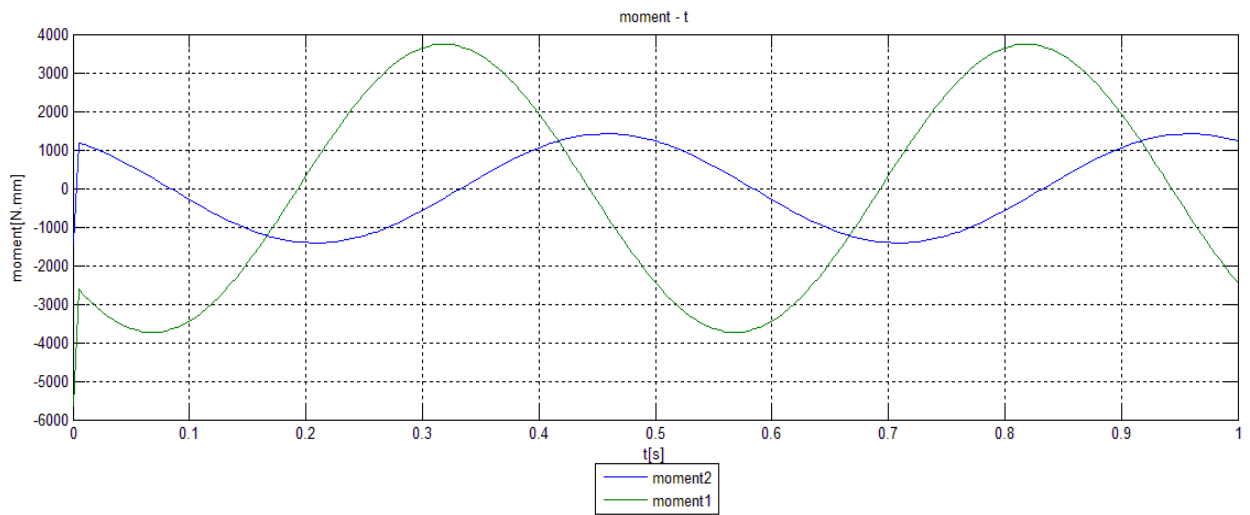


Obr. 7.9 – Schéma v Simulinku

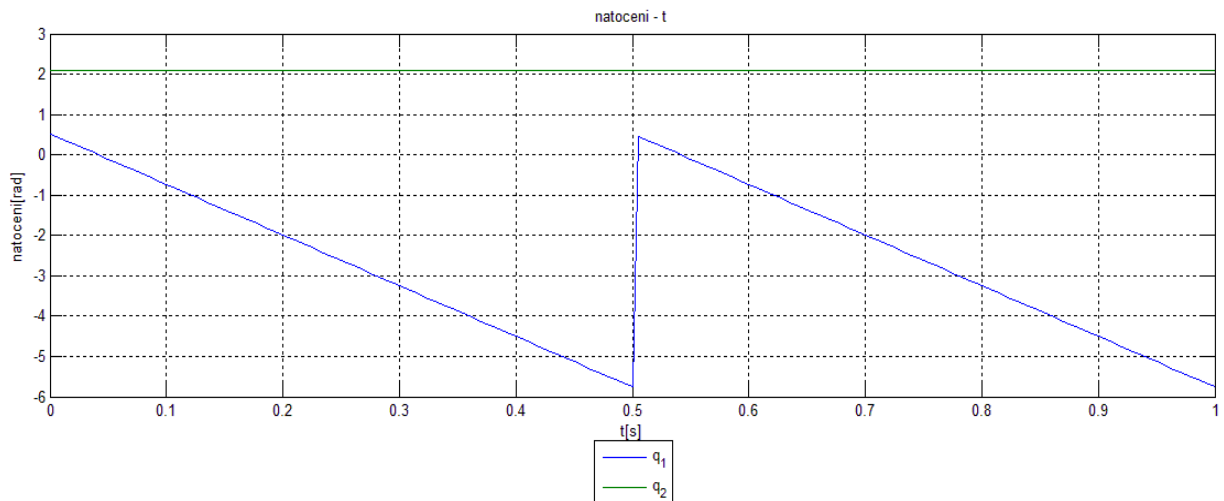
Postupně budeme simulovat všechny tři trajektorie, budeme vykreslovat polohu koncového efektoru (obr. 7.10, 7.13, 7.16), momenty (obr. 7.11, 7.14, 7.17) a zobecněné souřadnice (obr. 7.12, 7.15, 7.18).



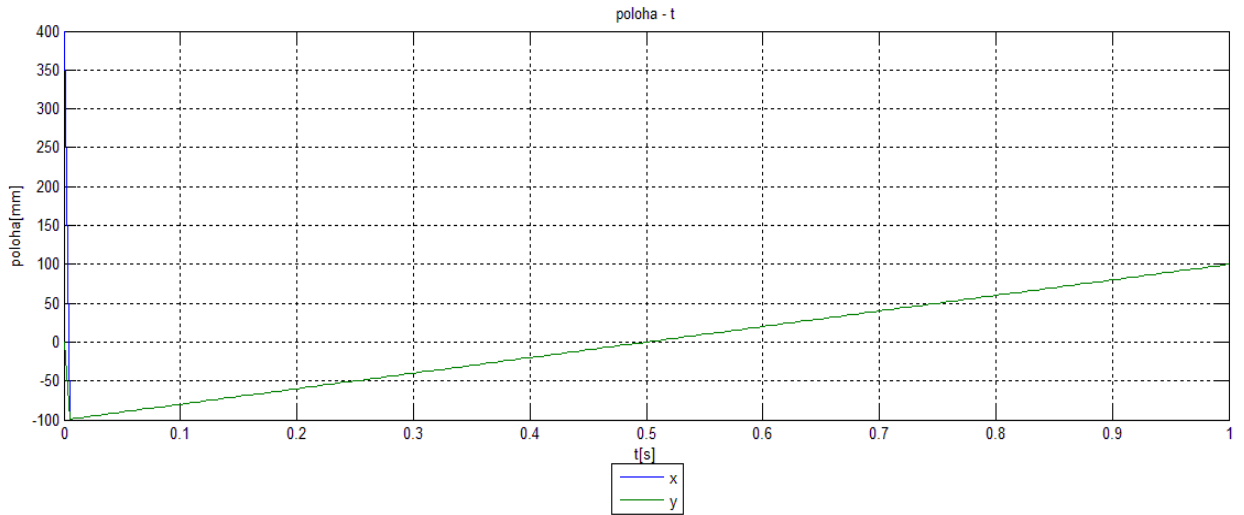
Obr. 7.10 – Kruhová trajektorie, závislost polohy na čase t



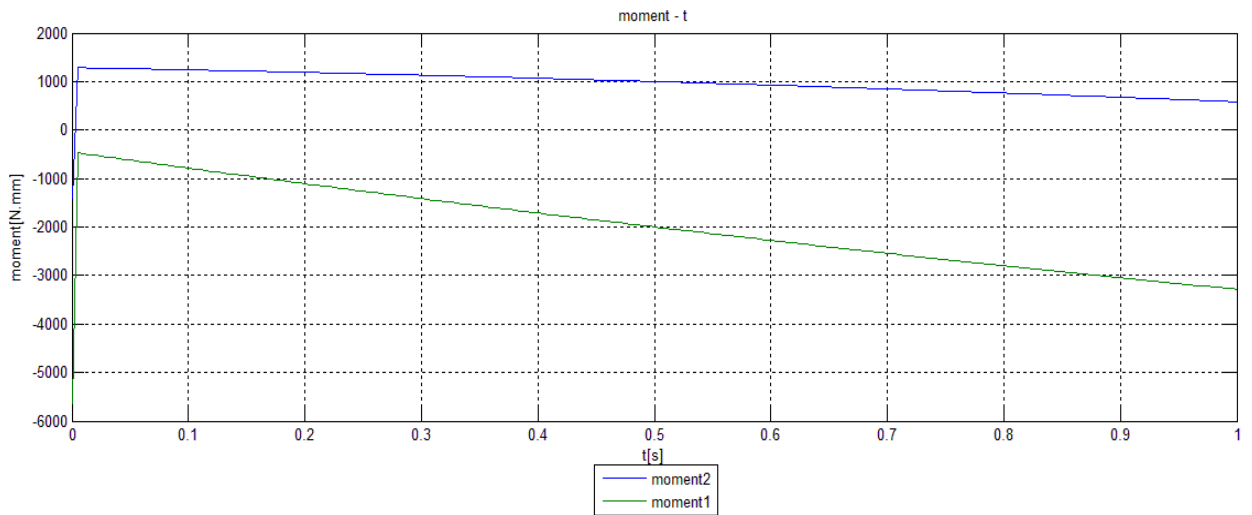
Obr. 7.11 – Kruhová trajektorie, závislost momentů na čase t



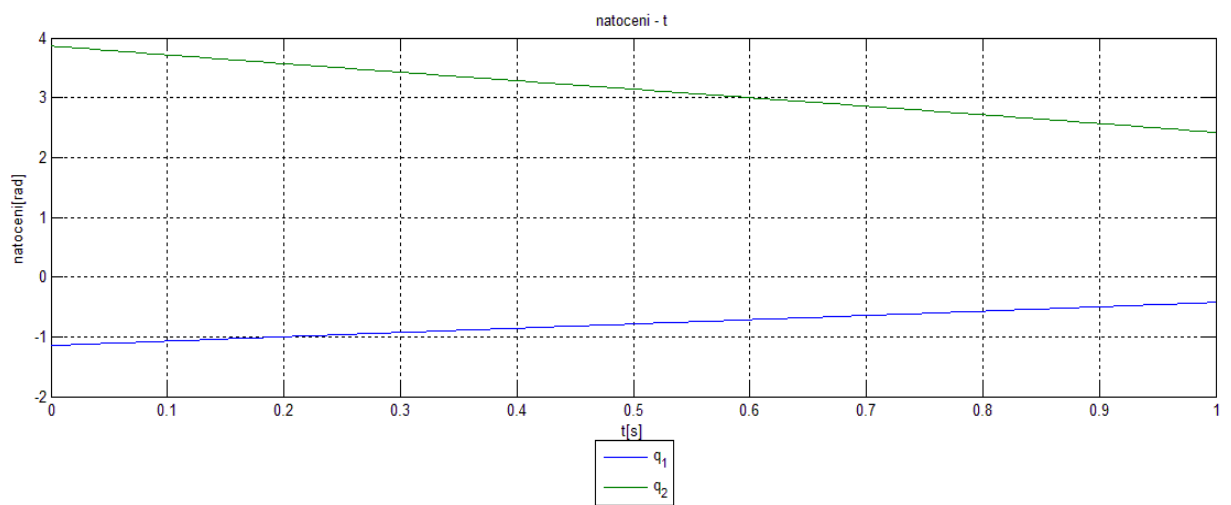
Obr. 7.12 – Kruhová trajektorie, závislost zobecněných souřadnic na čase t



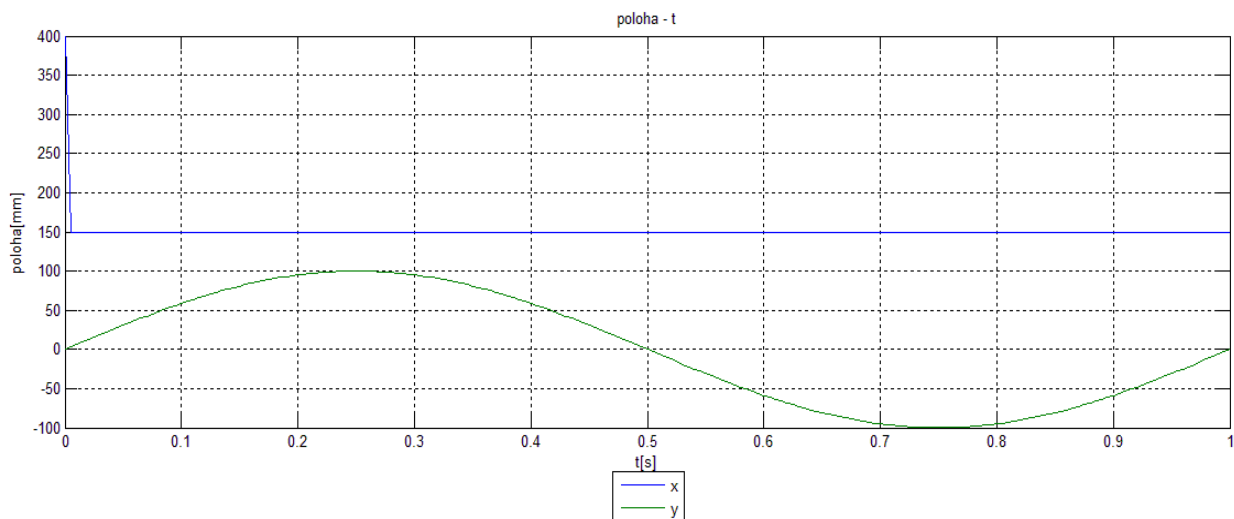
Obr. 7.13 – Přímá trajektorie, závislost polohy na čase t



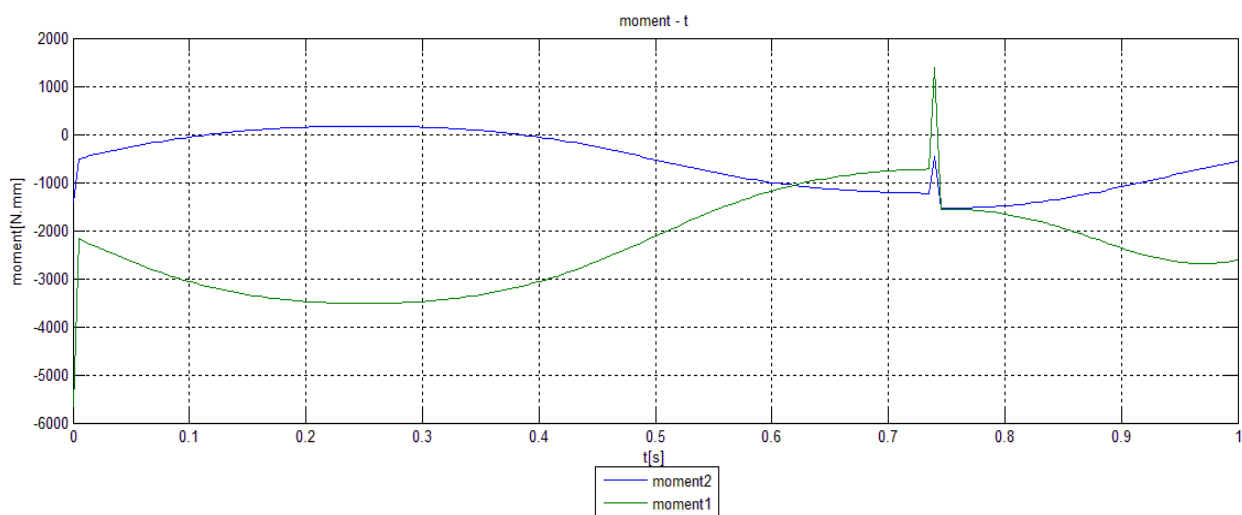
Obr. 7.14 – Přímá trajektorie, závislost momentů na čase t



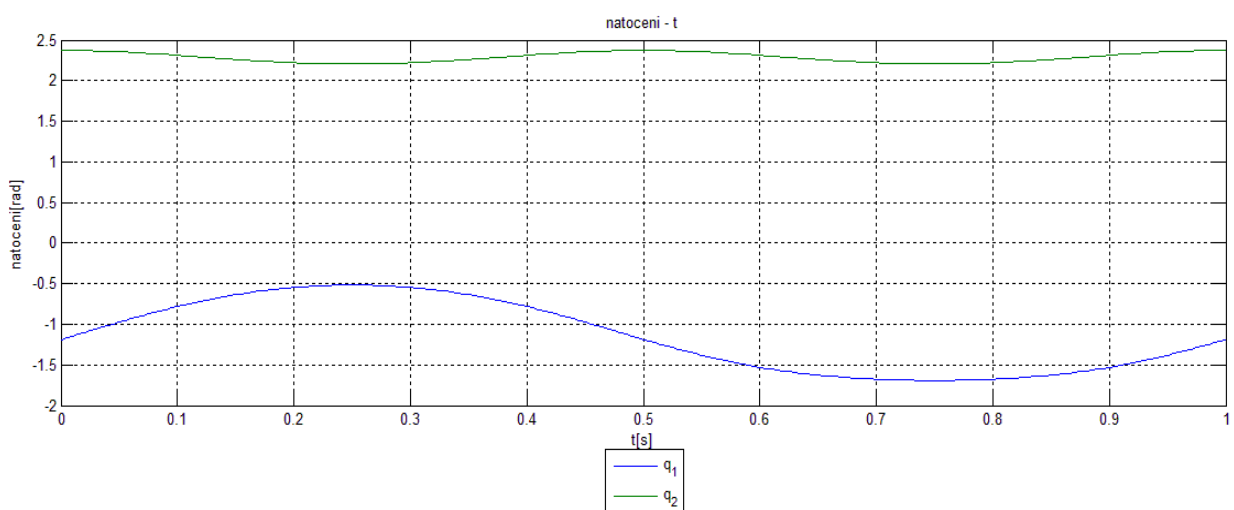
Obr. 7.15 – Přímá trajektorie, závislost zobecněných souřadnic na čase t



Obr. 7.16 – Kývavý pohyb, závislost polohy na čase t



Obr. 7.17 – Kývavý pohyb, závislost momentů na čase t



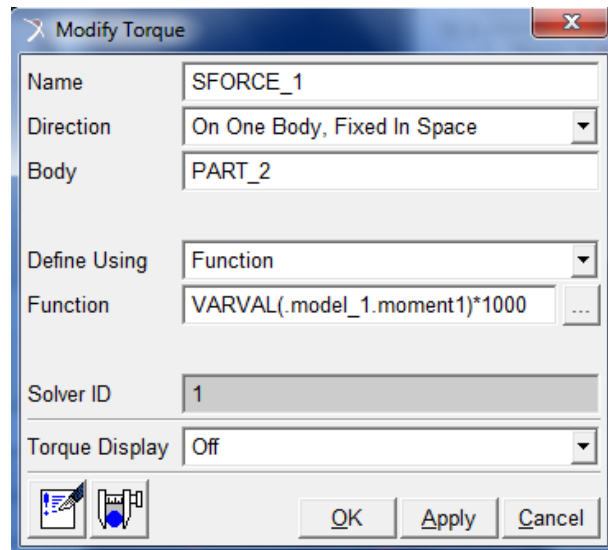
Obr. 7.18 – Kývavý pohyb, závislost zobecněných souřadných na čase t  
Výsledky co-simulace dále použijeme jako podklady pro návrh pohonu.

## Model manipulátoru s motory:

### Model v Adams:

Opět vytvoříme oba články a kin. vazby. Model se bude lišit pouze výstupními a vstupními proměnnými.

Vstupem budou momenty motorů. Stavové proměnné udávající jejich hodnotu, které později definujeme jako vstupy modelu, pojmenujeme moment1 a moment2. Samotné momenty vytvoříme ikonou Applied Force: Torque (Single – Component) a přiřadíme jim odpovídající hodnotu, např. moment motoru v první vazbě je na obr. 7.19:



Obr. 7.19 – Zadání vstupního momentu

Protože velikosti momentů požadujeme v jednotkách Nmm, násobíme velikost momentu tisícem (model v Simuliku udává moment v Nm, bude rozebrán později). Druhý moment modelujeme jako moment mezi dvěma tělesy, v řádku Direction tedy zvolíme Between Two Bodies. Jako Action Body (těleso, na něž působí předepsaný moment) nastavíme PART\_3 (druhý článek) a jako Reaction Body (těleso, na něž působí reakční moment) nastavíme PART\_2.

Výstupem budou souřadnice x, y a úhlové rychlosti otáčení článků kolem osy z. Vytvoříme pro ně stavové proměnné u\_rychlost1 a u\_rychlost2, jejich velikost bude určena funkcí WZ.

Při exportu nastavíme Adams/Solver Choice na FORTRAN, zbytek postupu je stejný.

### Model v Simulinku:

Nyní vybereme pohon podle momentů působících v kin. vazbách manipulátoru při simulacích v kapitole co – simulace. Při simulovaných trajektoriích působily momenty v řádu jednotek Nm. Pro návrh pohonu jsme si proto stanovili podmínku:

$$i_{12} \cdot \eta \cdot M_n \geq 10 N \cdot m \quad (36)$$

Kde  $i_{12}$  je převodový poměr převodovky,  $\eta$  je její účinnost a  $M_n$  je nominální moment motoru. V katalogu Maxon Motor jsme vybrali DC motor re50 370356 ( $M_n = 0,42$  Nm) s planetovou převodovkou GP42 203120 ( $i_{12} = 43$ ,  $\eta = 72$  %):

$$43 \cdot 0,72 \cdot 0,42 N \cdot m = 13,9 N \cdot m \geq 10 N \cdot m$$

Motor je popsán rovnicemi:

$$\begin{aligned} \frac{di}{dt} &= \frac{-R}{L} \cdot i(t) - \frac{K_b}{L} \cdot \omega(t) + \frac{1}{L} \cdot u(t) \\ \frac{d\omega}{dt} &= \frac{-1}{J} \cdot K_f \cdot \omega(t) + \frac{1}{J} \cdot K_m \cdot i(t) \end{aligned} \quad (37)$$

Kde R je el. odpor motoru, L je indukčnost motoru,  $K_m$  je momentová konstanta,  $K_b$  je napěťová konstanta,  $K_f$  je koeficient viskózního tření, J je moment setrvačnosti a U jmenovité vstupní napětí. Podle katalogu má motor tyto parametry:

$$R = 0,608 \ \Omega;$$

$$L = 0.423 \ \text{H};$$

$$K_m = 0,0934 \ \frac{\text{N} \cdot \text{m}}{\text{A}} \ ;$$

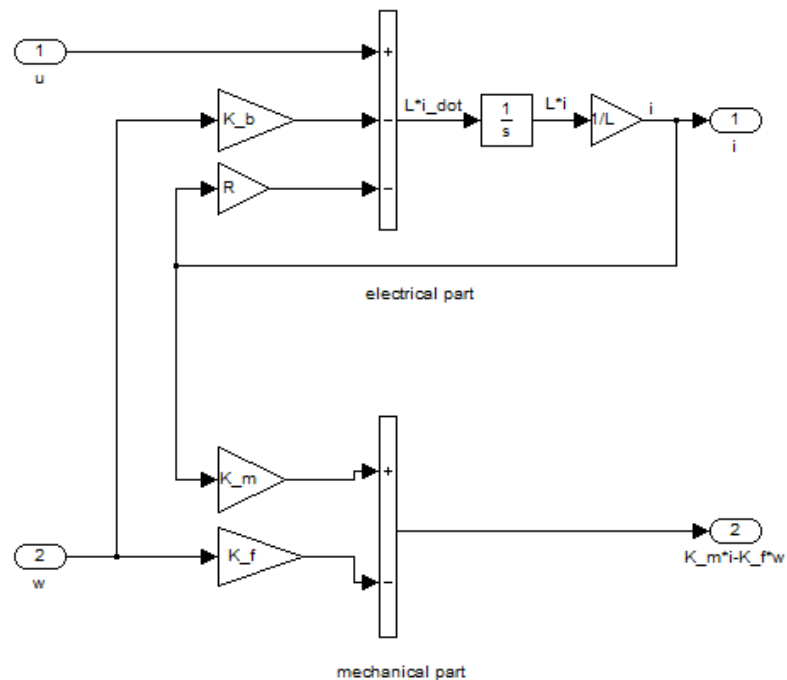
$$K_b = 0.0935 \ \frac{\text{V}}{\text{rad} \cdot \text{s}} \ ;$$

$$K_f = 0.0000048 \ \frac{\text{kg} \cdot \text{m}^3}{\text{s}} \ ;$$

$$J = 54,2\text{e-}6 \ \text{kg} \cdot \text{m}^3 \ ;$$

$$U = 48 \ \text{V}.$$

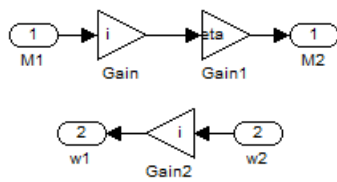
Model motoru byl vytvořen v Simulinku podle [14] (obr. 7.20):



Obr. 7.20 – Model DC motoru

Vstupem modelu je napětí a úhlová rychlost otáčení hřídele motoru v radiánech, výstupem je proud motoru a moment.

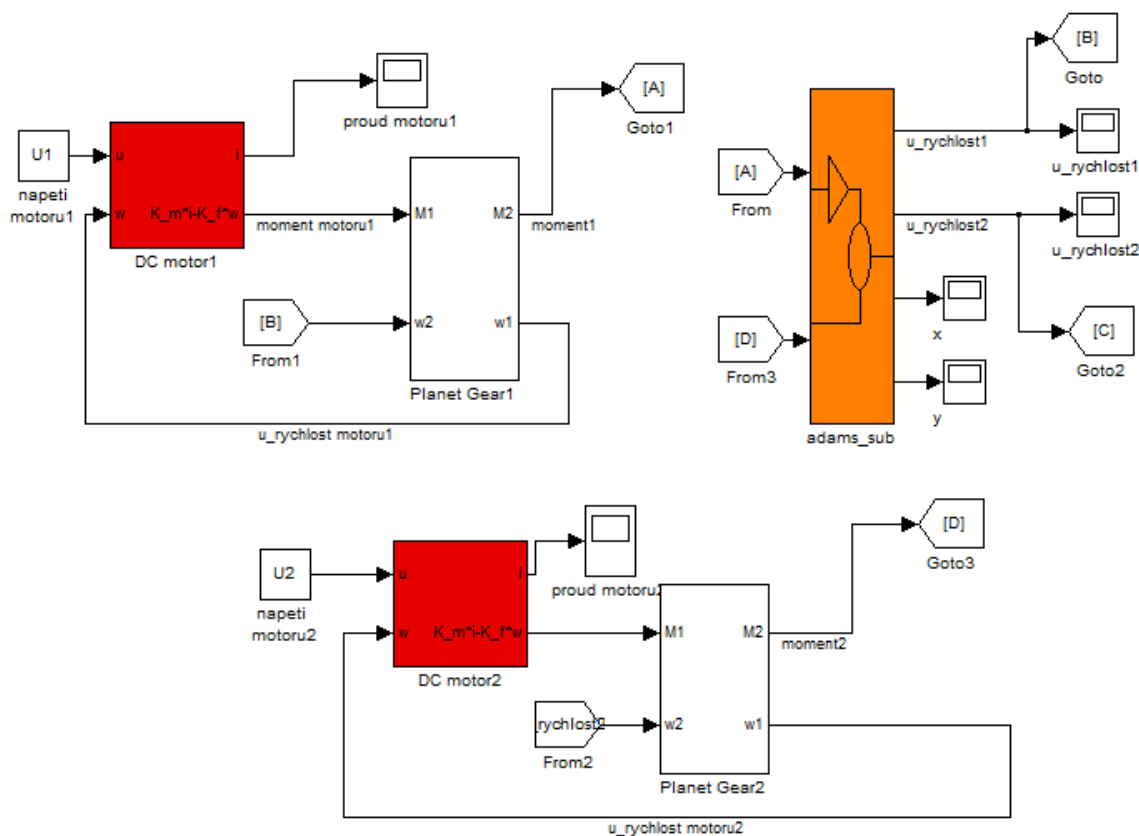
Planetová převodovka GP42 203120 má převodový poměr  $i_{12} = 43$  a účinnost  $\eta = 72 \%$ , její setrvačný moment zanedbáme, oproti setrvačným momentům v manipulátoru je malý. Model planetové převodovky v Simulinku ukazuje obr. 7.21:



Obr. 7.21 – Model planetové převodovky

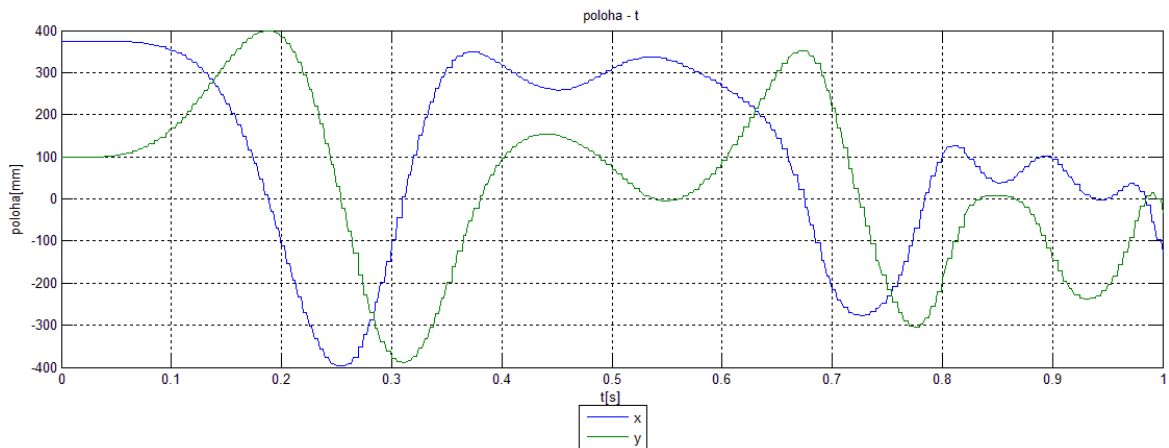
Parametry motoru a převodovky uložíme do m-file.

Vytvoříme model manipulátoru s pohony, tvořenými DC motorem a planetovou převodovkou (obr. 7.22):

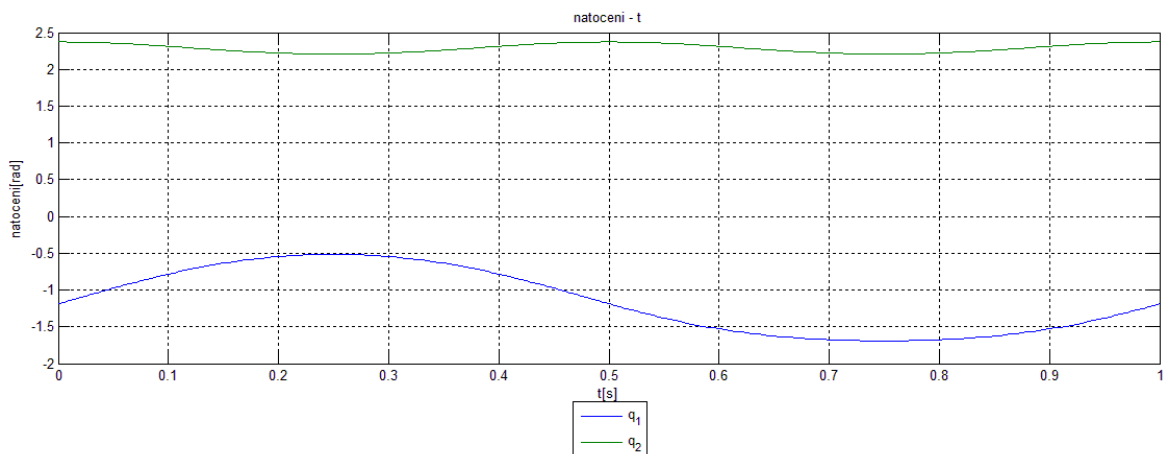


Obr. 7.22 – Model manipulátoru s pohony

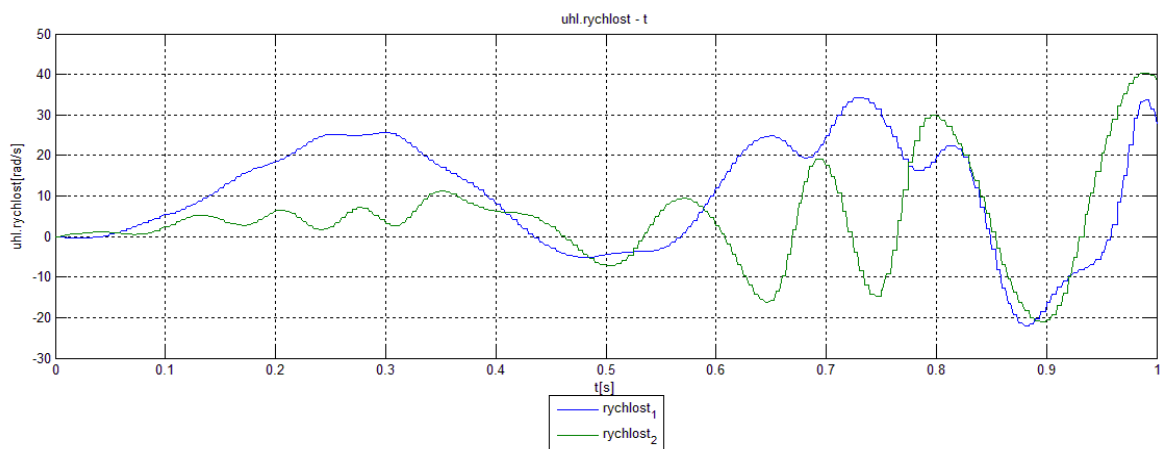
Na zvolených hodnotách  $U_1 = 45 \text{ V}$  a  $U_2 = 16 \text{ V}$  ukážeme, že model je funkční (obr. 7.23 – 7.26):



Obr. 7.23 – Závislost polohy na čase  $t$

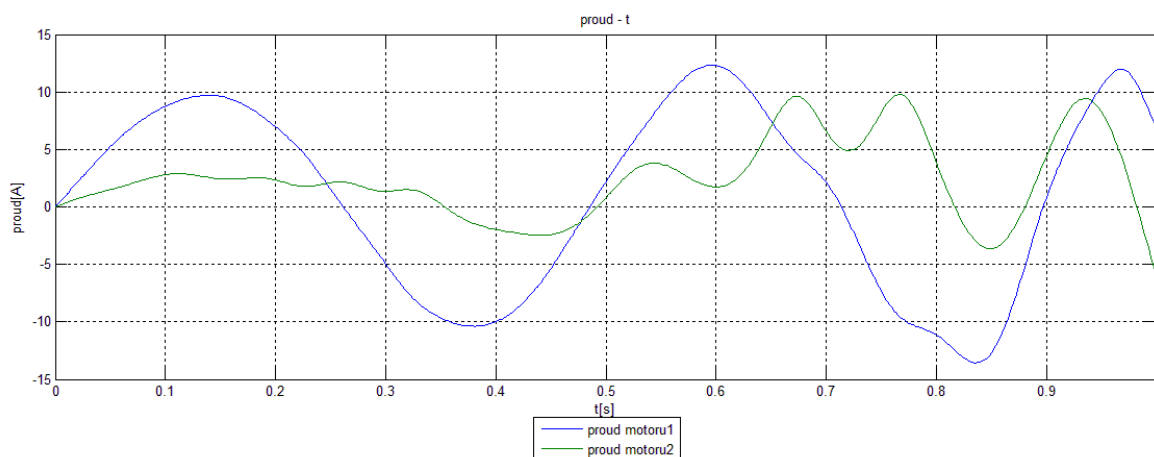


Obr. 7.24 – Závislost zobecněných souřadnic na čase  $t$



Obr. 7.25 – Závislost úhlové rychlosti na čase  $t$





Obr. 7.26 – Závislost proudů motorů na čase t

Vytvořený model manipulátoru s pohony tvořenými DC motorem a planetovou převodovkou je jen ukázkový, vhodný například k výuce. Při návrhu reálného manipulátoru bychom museli nejméně aplikovat řízení.

## **8. Závěr:**

Robotika je multidisciplinární vědní obor, který zasahuje především do mechaniky, elektroniky a IT technologií. Rešeršní část práce si kladla za cíl svou strukturou i rozsahem dát čtenáři přehled zejména o jejím využití v oblasti průmyslu. Práce se zabývá přímou kinematikou, metodami pro řešení inverzní úlohy kinematiky, metodami pro analýzu dynamiky. Problematice řízení byla věnována samostatná kapitola, na kterou navazuje popis základních druhů motorů a převodů v kapitole pohony.

Další část práce demonstruje základní kroky návrhu na příkladu jednoduchého rovinného manipulátoru. Pro řešení inverzní úlohy byla použita vektorová metoda. Správnost tohoto řešení pak byla ověřena v prostředí Simulink. Dále byl vytvořen multi-body model manipulátoru v MSC Adams, který byl následovně použit při analýze dynamiky, na základě které byly nakonec navrženy pohony s vyhovujícími parametry. Na příkladu byl také vysvětlen princip co-simulace nástrojů Adams – Simulink kombinující možnosti obou simulačních prostředí. Tento přístup byl v daném případě zvolen pro simulaci inverzní kinematiky manipulátoru v prostředí Simulink v kombinaci s multi-body modelem dynamiky v prostředí Adams.

Hlavním přínosem práce je zejména možnost jejího dalšího využití v rámci výuky předmětů na oboru Mechatronika. Studenti se tak mohou seznámit se základní problematikou návrhu průmyslového manipulátoru a obecně s modelováním mechatrických soustav.

Závěrem lze konstatovat, že vytyčené cíle práce byly beze zbytku splněny.

### Seznam použité literatury:

- [1] SKAŘUPA, J. *Průmyslové roboty a manipulátory*. Ostrava: Ediční středisko VŠB – TUO, 2007. 260 s. ISBN 978-80-248-1522-0. Dostupné z: [http://www.elearn.vsb.cz/archivcd/FS/PRM/Text/Skripta\\_PRaM.pdf](http://www.elearn.vsb.cz/archivcd/FS/PRM/Text/Skripta_PRaM.pdf)
- [2] JAZAR, R.N. *Theory of Applied Robotics: Kinematics, Dynamics, and Control*. Springer, 2007. 693 s. ISBN 978-0-387-68964-7.
- [3] SPONG, M.W., Seth HUTCHINSON, M. VIDYASAGAR. *Robot Modeling and Control*. John Wiley and Sons, 2007. 407 s.
- [4] SKALICKÝ, J. *Teorie řízení I*. Brno: VUT FEKT ÚVEE Brno, 2002. 98 s. ISBN: 80-214-2112-6.
- [5] PEŇÁZOVÁ, Milena. Když se řekne... Robot. *Automatizace*[online], ročník 51, č. 7-8, 488 s. [cit. 2012-04-10]. Dostupné na WWW: <http://www.automatizace.cz/article.php?a=2194>
- [6] LIU X.-J., J. WANG, G. PRITSCHOW. A new family of spatial 3-DoF fully-parallel manipulators with high rotational capability. *Mechanism and Machine Theory*. Elsevier, 2005. ročník 40, č. 4, s. 477
- [7] SICILIANO, Bruno, Lorenzo SCIAVICCO, Luigi VILLANI a Giuseppe ORIOLO. *Robotics: Modelling, Planning and Control*. Springer, 2010. 632 s. ISBN 978-1-84628-642-1.
- [8] MOSTÝN, Vladimír, SKAŘUPA, Jiří. *Teorie průmyslových robotů*. 1. vyd. Košice: Edícia vedeckej a odbornej literatúry – Strojnícka fakulta TU v Košiciach, VIENALA Košice, 2000. 150 s. ISBN 80-88922-35-6.
- [9] BINNEY, J.J.. *Classical Mechanics*. University of Oxford, 2006, 29 s. [cit. 2012-04-25] Dostupné na WWW: <http://www-thphys.physics.ox.ac.uk/people/JamesBinney/cmech.pdf>
- [10] Lineární diferenciální rovnice. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2012-04-15]. Dostupné na WWW: [http://cs.wikipedia.org/wiki/Line%C3%A1rn%C3%AD\\_diferenci%C3%A1ln%C3%AD\\_rovnice](http://cs.wikipedia.org/wiki/Line%C3%A1rn%C3%AD_diferenci%C3%A1ln%C3%AD_rovnice)
- [11] MURRAY, R.M., Zexiang LI a S.S. SASTRY. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994, s. 179-189. Dostupné na WWW: <http://www.cds.caltech.edu/~murray/books/MLS/pdf/mls94-complete.pdf>
- [12] VAVŘIŇÁK, Petr. *Elektrické stroje a přístroje*. 2006, s. 51-76. Dostupné na WWW: [http://www.sse-najizdarne.cz/dokumenty/studijni\\_materialy/elektricke\\_stroje\\_a\\_pristroje.pdf](http://www.sse-najizdarne.cz/dokumenty/studijni_materialy/elektricke_stroje_a_pristroje.pdf)

[13] AKBARZADEH, Alireza a Javad ENFERADI. *A Virtual Work Based Algorithm for Solving Direct Dynamics Problem of a 3-RRP Spherical Parallel Manipulator*. Springer, 2010, s. 40. DOI 10.1007/s10846-010-9469-9.

[14] BŘEZINA, L. *Optimization of a parallel mechanism design with respect to a Stewart platform control design*. Brno: Brno University of Technology, Faculty of Mechanical Engineering, 2010, s. 33 – 34.

### Seznam použitých symbolů:

- $q_i$  ... zobecněná souřadnice  $i$  – tého článku,  
 $\mathbf{q}$  ... vektor zobecněných souřadnic,  
 $\mathbf{p}$  ... vektor polohy nástroje,  
 $\mathbf{o}$  ... vektor orientace nástroje,  
 $\mathbf{w}$  ... komplexní vektor polohy nástroje,  
 $\mathbf{i}$  ... jednotkový vektor ve směru osy  $x$ ,  
 $\mathbf{j}$  ... jednotkový vektor ve směru osy  $y$ ,  
 $\mathbf{k}$  ... jednotkový vektor ve směru osy  $z$ ,  
 $\mathbf{u}, \mathbf{v}$  ... dva různoběžné vektory,  
 $\alpha$  ... úhel mezi vektory  $\mathbf{u}$  a  $\mathbf{v}$ ,  
 $\mathbf{R}$  ... transformační matice pro natočení,  
 $\mathbf{A}_{i-1,i}$  ... univerzální transformační matice  
 $\vartheta_i$  ... úhel mezi osami  $x_{i-1}$  a  $x_i$  při otáčení kolem  $z_{i-1}$ ,  
 $d_i$  ... nejkratší vzdálenost (normála) mezi osami  $x_{i-1}$  a  $x_i$ , kladný směr ve směru  $z_{i-1}$ ,  
 $a_i$  ... nejkratší vzdálenost (normála) mezi osami  $z_{i-1}$  a  $z_i$ , kladný směr ve směru  $x_i$ ,  
 $\alpha_i$  ... úhel mezi osami  $z_{i-1}$  a  $z_i$  při otáčení kolem  $x_i$ ,  
 $\mathbf{T}_{bn}$  ... celková transformační matice,  
 $\mathbf{J}$  ... jakobián,  
 $\Delta \mathbf{w}$  ... změna komplexního vektoru polohy,  
 $\mathbf{J}^{-1}$  ... inverze jakobiánu,  
 $\mathbf{J}^+$  ... pseudoinverze jakobiánu,  
 $\mathbf{J}^T$  ... transformace jakobiánu,  
 $E$  ... chyba polohování,  
 $\Delta \mathbf{p}$  ... chyba polohy,  
 $\Delta \mathbf{o}$  ... chyba orientace,  
 $l_i$  ... délka  $i$  - tého článku manipulátoru,  
 $i_v$  ... počet stupňů volnosti v uvažovaném prostoru,  
 $n_t$  ... počet těles (včetně základního),  
 $n_v$  ... počet vazeb,  
 $\xi_i$  ... počet stupňů volnosti odebraný  $i$  - tou vazbou,  
 $i$  ... počet stupňů volnosti mechanismu,  
 $x, y$  ... souřadnice koncového efektoru v globálním souřadném systému,  
 $x_b, y_b$  ... souřadnice koncového efektoru v globálním souřadném systému (indexování použito pro odlišení od vstupních hodnot v simulaci),  
 $\alpha, \beta, \gamma$  ... pomocné úhly,  
 $i_{12}$  ... převodový poměr převodovky,  
 $\eta$  ... účinnost převodovky,  
 $M_n$  ... nominální moment motoru,  
 $R$  ... el. odpor motoru,  
 $L$  ... indukčnost motoru,  
 $K_m$  ... momentová konstanta motoru,  
 $K_b$  ... napěťová konstanta motoru,  
 $K_f$  ... koeficient viskózního tření,  
 $J$  ... moment setrvačnosti motoru,  
 $U$  ... jmenovité vstupní napětí motoru,  
 $U_1$  ... napájecí napětí DC motoru1,  
 $U_2$  ... napájecí napětí DC motoru2

**Seznam příloh:**

Příloha 1 – Dopředná kinematika v Matlabu

Příloha 2 – Inverzní kinematika v Matlabu

## **Příloha 1: Dopředná kinematika v Matlabu**

**Funkce pro výpočet  $x_b$ :**

```
function[x_b]=foward_x(q1,q2)
A_1=[cos(q1),-sin(q1),0,200*cos(q1);
sin(q1),cos(q1),0,200*sin(q1);0,0,1,0;0,0,0,1];
A_2=[cos(q2),-sin(q2),0,200*cos(q2);
sin(q2),cos(q2),0,200*sin(q2);0,0,1,0;0,0,0,1];
T=A_1*A_2;
x_b=T(1,4);
```

**Funkce pro výpočet  $y_b$ :**

```
function[y_b]=foward_y(q1,q2)
A_1=[cos(q1),-sin(q1),0,200*cos(q1);
sin(q1),cos(q1),0,200*sin(q1);0,0,1,0;0,0,0,1];
A_2=[cos(q2),-sin(q2),0,200*cos(q2);
sin(q2),cos(q2),0,200*sin(q2);0,0,1,0;0,0,0,1];
T=A_1*A_2;
y_b=T(2,4);
```

## Příloha 2: Inverzní kinematika v Matlabu

Funkce pro výpočet  $q_1$ :

```
function[q1] = manipulator1(x,y,q0);
l1=200;
l2=200;
if x==0 & y==0
    q1=q0;
elseif x<0
    i=atan(y/x)+acos((l1^2-l2^2+(x^2+y^2))/(2*l1*sqrt(x^2+y^2)))-pi;
    k=(atan(y/x)-acos((l1^2-l2^2+(x^2+y^2))/(2*l1*sqrt(x^2+y^2)))-pi);
    switch abs(i-q0)<abs(k-q0);
        case 1
            q1=(atan(y/x)+acos((l1^2-l2^2+(x^2+y^2))/(
(2*l1*sqrt(x^2+y^2)))-pi));
        case 0
            q1=(atan(y/x)-acos((l1^2-l2^2+(x^2+y^2))/(
(2*l1*sqrt(x^2+y^2)))-pi));
        end
    else
        i=atan(y/x)+acos((l1^2-l2^2+(x^2+y^2))/(2*l1*sqrt(x^2+y^2)));
        k=(atan(y/x)-acos((l1^2-l2^2+(x^2+y^2))/(2*l1*sqrt(x^2+y^2)));
        switch abs(i-q0)<abs(k-q0);
            case 1
                q1=(atan(y/x)+acos((l1^2-l2^2+(x^2+y^2))/(
(2*l1*sqrt(x^2+y^2))));
            case 0
                q1=(atan(y/x)-acos((l1^2-l2^2+(x^2+y^2))/(
(2*l1*sqrt(x^2+y^2))));
            end
        end
    end
```

Funkce pro výpočet  $q_2$ :

```
function[q2] = manipulator2(x,y,q0);
l1=200;
l2=200;
if x==0 & y==0
    q2= pi;
elseif x<0
    i=-pi+atan(y/x)+acos((l1^2-l2^2+(x^2+y^2))/(2*l1*sqrt(x^2+y^2)));
    k=-pi+atan(y/x)-acos((l1^2-l2^2+(x^2+y^2))/(2*l1*sqrt(x^2+y^2)));
    switch abs((i)-(q0))<abs((k)-(q0));
        case 0
            q2=pi-acos((l2^2+l1^2-(x^2+y^2))/(2*l1*l2));
        case 1
            q2=pi+acos((l2^2+l1^2-(x^2+y^2))/(2*l1*l2));
        end
    else
        i=atan(y/x)+acos((l1^2-l2^2+(x^2+y^2))/(2*l1*sqrt(x^2+y^2)));
        k=(atan(y/x)-acos((l1^2-l2^2+(x^2+y^2))/(2*l1*sqrt(x^2+y^2)));
        switch abs((i)-(q0))<abs((k)-(q0));
            case 0
                q2=pi-acos((l2^2+l1^2-(x^2+y^2))/(2*l1*l2));
            case 1
                q2=pi+acos((l2^2+l1^2-(x^2+y^2))/(2*l1*l2));
            end
        end
    end
```