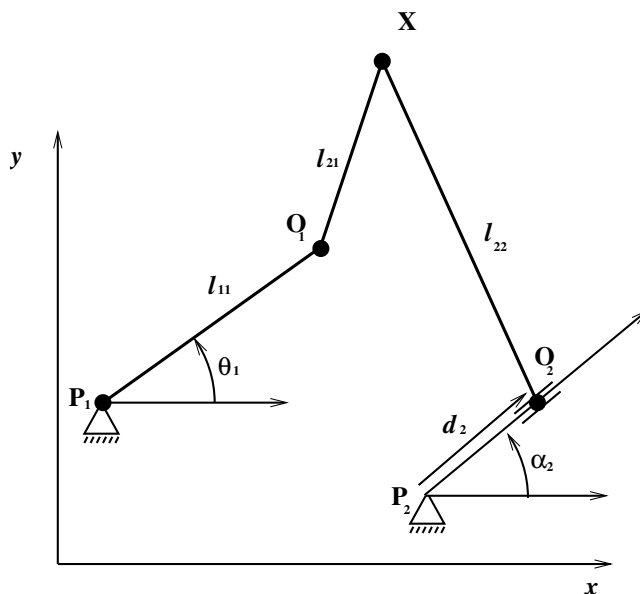


## Úloha 1

**Název:** Kinematika paralelního planárního manipulátoru.

**Zadání:**

Máte zadán paralelní planární manipulátor. Manipulátor se obecně skládá ze čtyřech ramen spojených třemi rotačními nepoháněnými klouby. Manipulátor je vybaven dvěma pohony, které jsou pevně svázány se světovou souřadnicovou soustavou v upevňovacích bodech  $P_1$  a  $P_2$ . Každý z pohonů může být buď posuvný nebo rotační. Kombinace pohonů tvoří manipulátory s různou strukturou. Příklad manipulátoru s jedním rotačním a jedním posuvným kloubem je na obrázku 1.



Obrázek 1: Paralelní planární manipulátor s jedním rotačním a jedním posuvným kloubem.

Světová souřadnicová soustava je označena  $x, y$ . Koncový bod je označen  $X$ . Kloubové souřadnice jsou označeny pro rotační klouby  $\theta_i$  a pro posuvné klouby  $d_i$ . Délky ramen jsou označeny  $l_{ij}$  s indexy odpovídajícími pořadí ramen. Směr pohybu posuvného kloubu je dán sklonem vůči světové souřadnicové soustavě, který je definován úhlem  $\alpha_i$ . Všechny úhly jsou měřeny v kladném směru (naznačeno šipkou).

Máte zadány tři varianty úlohy: lehká, střední, těžká. Liší se druhem použitých kloubů. Větší rozmanitost a více otočných kloubů typicky znamená obtížnější řešení inverzní kinematické úlohy. Vy si můžete vybrat jakoukoliv variantu. Za těžší variantu je více bodů. Dávejte si pozor, abyste si nepopletli obrázky a

data jednotlivých variant.

Součástí zadání je náčrtek Vaší varianty manipulátoru s popisem. Fixní parametry manipulátoru jsou značeny modře, kloubové souřadnice jsou červeně. Kóty úhlů a rozměrů, které jsou nulové, nejsou označeny a kótovány.

### Úkoly:

1. Pro každého studenta je připraven jiný robot. Prostudujte zadaný manipulátor.
2. Sestavte příslušné vazbové rovnice.
3. Vyřešte přímou kinematickou úlohu.
4. Vyřešte inverzní kinematickou úlohu manipulátoru.
5. Obě řešení doprovodte obrázky a stručným popisem.
6. Obě řešení implementujte v Matlabu.
7. Obě řešení vyzkoušejte na Vámi vygenerovaných datech.
8. Zkontrolujte syntaxi Vašeho kódu příkazem `mlint`. Váš kód by neměl generovat žádné chyby ani varovná hlášení.
9. Nezapomeňte přibalit soubor **variant.m** s vybranou variantou.
10. Připravte Vaše odevzdávané funkce v Matlabu.
11. Před nahráním výsledků do odevzdávacího systému je nutné vše zabalit do archivu typu ZIP tak, aby odevzdávané funkce byly přímo v kořeni archivu.
12. Výsledný archiv nahrajte do odevzdávacího systému.
13. Průvodní nákresy, schemata či odvození případně zprávu připravte ve formátu PDF. Zprávu nahrajte jako výsledek úlohy označené stejným pořadovým číslem jako zadaná úloha s dovětkem "report".

### Obsah zprávy:

1. Formality (předmět a název zprávy, jméno, username, datum, případně verze zprávy)
2. Formulace řešeného problému. Popište matematicky, co je vstupem a výstupem (nezapomeňte, že úloha má dva podúkoly). Použijte matematickou, ne programátorskou notaci.
3. Obrázek zadaného manipulátoru spolu s vyznačenými geometrickými tvary, které byly použity při výpočtu (např. kružnice, přímky,...). Popisek obrázku musí vysvětlovat všechno, co se na obrázku nachází. Pro obrázky použijte pokud možno vektorový formát.

4. Vazbové rovnice explicitně napsané.
5. Popis výpočtu DKT a IKT:
  - (a) Uvedte všechny rovnice potřebné k implementaci kódu včetně rovnic, které popisují výpočet průsečíku. Všechny proměnné a symboly musí být definovány před nebo hned za rovnicí, ve které jsou použity.
  - (b) Explicitně napište rovnice pro výpočet požadovaných hodnot. Např. Souřadnice  $x = \dots$
  - (c) Diskutujte počet řešení.

### Formát funkcí a testovacích dat:

- V souboru ve formátu MAT je uvedena proměnná **robot** popisující manipulátor.
- Proměnná **robot** je struktura ve tvaru:
  - **struct**, dvě čísla označující typ kloubů (1 – posuvný, 2 – rotační).
  - **rotations**, dvě čísla, která udávají úhel natočení kloubu  $\alpha_i$  v radiánech. V případě rotačního kloubu je uvedena hodnota NaN.

$$rotations = [\alpha_1, \alpha_2]$$

- **links**, čtyři čísla, která jsou délkou  $l_{ij}$  ramen v jednotkách délky. V případě posuvného kloubu vždy jedno rameno schází. V proměnné **links** je pro rameno ( $l_{11}$  nebo  $l_{12}$ ) uvedena hodnota NaN tam, kde je použit posuvný poháněný kloub. Délky ramen příslušející jednotlivým cestám od pevného uchycení do koncového bodu jsou uloženy jako sloupcové vektory.

$$links = \begin{bmatrix} l_{11} & l_{12} \\ l_{21} & l_{22} \end{bmatrix}$$

- **origins**, čtyři čísla, která definují polohu upevňovacích bodů (kloubů)  $P_1, P_2$  ve světové souřadné soustavě. Počátky jsou v této proměnné uloženy jako sloupcové vektory.

$$origins = \begin{bmatrix} x_{P1} & x_{P2} \\ y_{P1} & y_{P2} \end{bmatrix}$$

Úhly  $\alpha_i$  a délky ramena  $l_{ij}$  jsou na obrázcích v příslušném měřítku, délky posuvných a úhly otočných kloubů jsou jen naznačeny.

- Přímá kinematická úloha je funkce ve tvaru **XC=mydkt(robot,J)**, kde
  - **robot** je Vám zadaná proměnná popisující manipulátor ve tvaru, který je v souboru ve formátu MAT.

- **J** je matice, kde každý sloupec délky 2 reprezentuje kloubové souřadnice v pořadí kloubů dle struktury manipulátoru. Každý sloupec tedy představuje jednu polohu pohonů manipulátoru.
- **XC** je cell array, které obsahuje v jednotlivých buňkách různá řešení polohy koncového bodu  $X$  definované zadanými kloubovými souřadnicemi. **XC2(:,i)** je tedy druhé řešení (poloha koncového bodu) pro vstupní vektor kloubových souřadnic **J(:,i)**.
- Inverzní kinematická úloha je funkce ve tvaru **JC=myikt(robot,X)**, kde
  - **X** je matice, kde každý sloupec délky 2 reprezentuje souřadnice koncového bodu  $X$ . Každý sloupec tedy definuje jednu požadovanou polohu koncového bodu.
  - **JC** je cell array, která obsahuje v jednotlivých buňkách různé konfigurace řešení, zbylé parametry jsou stejné jako v případě přímé kinematické úlohy. **JC2(:,i)** je tedy druhé řešení (vektor kloubových souřadnic) pro požadovanou polohu koncového bodu **X(:,i)**.
- Zanedbejte periodická řešení typu  $\theta_i = \gamma + 2k\pi$ , taková řešení uvádějte jen v intervalu  $(-\pi, \pi]$ . Jestliže existuje nekonečně mnoho řešení, vraťte jedno (správné) řešení.
- Inverzní kinematická úloha může mít více řešení, vraťte je jako cell array více matic **JC** = { $J_1, J_2, \dots$ }, cell array **JC** bude mít vždy délku maximálního počtu řešení (konfigurací). To znamená, že má-li IKT dvě řešení pro např. 30. sloupec zadání, bude jedno řešení v 30. sloupci matice **JC{1}** a druhé řešení v 30. sloupci matice **JC{2}**. Pokud tedy robot může mít maximálně 4 neperiodická řešení, budete vracet vždy cell array délky 4. Ve sloupcích, kde robot má například jen 2 řešení, budou v **J{3}** a **J{4}** sloupce NaN.
- Pro paralelní manipulátory může mít i přímá kinematická úloha více řešení, vraťte je jako cell array více matic **XC** = { $X_1, X_2, \dots$ }.
- V singulárních konfiguracích, kdy robot má nekonečně nespočetně mnoho řešení typu  $\theta_1 = \delta_1 + \omega$  a  $\theta_2 = \delta_2 - \omega$ , kde  $\omega$  je jakékoliv reálné číslo, vraťte jen jedno řešení, pro např.  $\omega = 0$ .
- Všechny úhly jsou zadány v radiánech, vraťte také hodnoty v radiánech.
- Pokud je poloha nedosažitelná, vraťte vektory NaN v odpovídajících sloupcích matic **JC{i}**.
- Obrázek s příponou “fig” umožňuje natáhnout schéma manipulátoru do Matlabu a dokreslit další poznámky přímo do něho (ale můžete pracovat i jinak).

- Vaším zadáním je řešit konkrétního robota, takže např. v případě, že nějaké rameno je připojeno pod úhlem  $90^\circ$ , je možné tuto informaci využít. Vaše funkce ale i tak musí alespoň formálně akceptovat vstupní proměnnou **robot**.
- Vaše funkce nesmí psát na terminál, na disk, kreslit grafy ani generovat chyby. Vaše funkce musí pouze vracet výstupy v definovaných výstupních proměnných, a to pouze reálné hodnoty nebo NaN.
- Vaše funkce nesmí používat symbolický toolbox. Tyto funkce musí být v principu použitelné pro online řízení robotu a musí tedy zaručit, aby doba výpočtu nepřekročila zvolenou hodnotu odezvy systému. V každém případě není rozumné řešit symbolicky rovnice pro každou polohu robotu.
- Testovací data nejsou součástí zadání. Návrh testovacích dat (procedury) je standardní součástí inženýrského řešení, a proto musí být procvičováno.
- Je třeba sdělit systému, jakou úlohu jste vyřešili. To provedete připojením souboru `variant.m` obsahujícím funkci, jejíž výstupem bude jediná proměnná obsahující jediný znak, a to znak identifikující vybranou variantu. Znak “a” kóduje lehkou variantu, znak “b” střední variantu a znak “c” těžkou variantu. Vyřešení více variant jedné úlohy není dovoleno.

## Hodnocení:

Za tuto úlohu můžete získat maximálně 4 bodů.

- přímá kinematická úloha: 2 body,
- inverzní kinematická úloha v závislosti na obtížnosti: 1-2 body,
- pozdní odevzdání: 1 body dolu.