



Robotika: Inverzní Kinematika

Vladimír Petřík

vladimir.petrik@cvut.cz

14.10.2023

Kinematické úlohy

- ▶ Přímá kinematika (FK)
 - ▶ jak vypočítat pozici chapadla z konfigurace
 - ▶ $x = f_{fk}(q)$
 - ▶ x je vyjádřena v úkolovém prostoru, tzn. $SE(2)$, $SE(3)$, nebo $\mathbb{R}^2$, $\mathbb{R}^3$ pouze pro pozici
 - ▶ $q \in \mathbb{R}^N$ je konfigurace (prostor kloubu)



Kinematické úlohy

- ▶ Přímá kinematika (FK)
 - ▶ jak vypočítat pozici chapadla z konfigurace
 - ▶ $x = f_{fk}(q)$
 - ▶ x je vyjádřena v úkolovém prostoru, tzn. $SE(2)$, $SE(3)$, nebo $\mathbb{R}^2$, $\mathbb{R}^3$ pouze pro pozici
 - ▶ $q \in \mathbb{R}^N$ je konfigurace (prostor kloubu)
- ▶ Diferenciální kinematika
 - ▶ dává do vztahu rychlost chapadla s rychlostí kloubů
 - ▶ $\dot{x} = J(q)\dot{q}$



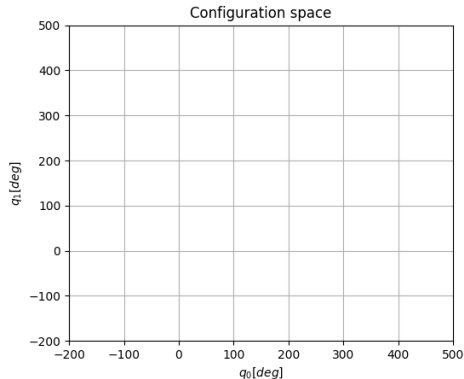
Kinematické úlohy

- ▶ Přímá kinematika (FK)
 - ▶ jak vypočítat pozici chapadla z konfigurace
 - ▶ $\boldsymbol{x} = f_{fk}(\boldsymbol{q})$
 - ▶ $\boldsymbol{x}$ je vyjádřena v úkolovém prostoru, tzn. $SE(2)$, $SE(3)$, nebo $\mathbb{R}^2$, $\mathbb{R}^3$ pouze pro pozici
 - ▶ $\boldsymbol{q} \in \mathbb{R}^N$ je konfigurace (prostor kloubu)
- ▶ Diferenciální kinematika
 - ▶ dává do vztahu rychlost chapadla s rychlostí kloubů
 - ▶ $\dot{\boldsymbol{x}} = J(\boldsymbol{q})\dot{\boldsymbol{q}}$
- ▶ Inverzní kinematika (IK)
 - ▶ jak vypočítat konfigurace robota pro danou konfiguraci chapadla
 - ▶ $\boldsymbol{q} \in f_{ik}(\boldsymbol{x})$



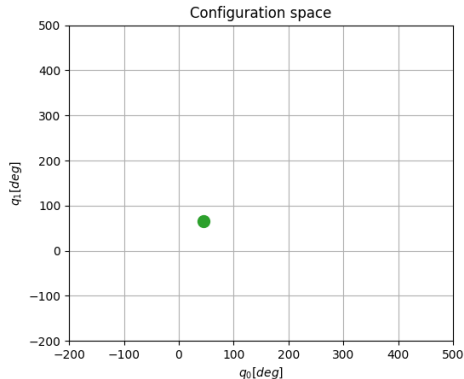
Příklad: Analytická IK pro strukturu RR

- ▶ Úkolový prostor: translace chapadla, $\mathbf{x} \in \mathbb{R}^2$
- ▶ Konfigurační (kloubový) prostor: $\mathbf{q} \in \mathbb{R}^2$



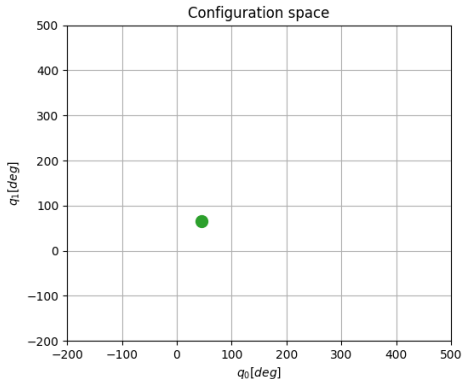
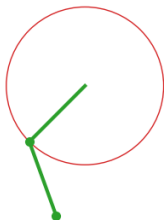
Příklad: Analytická IK pro strukturu RR

- ▶ Úkolový prostor: translace chapadla, $\mathbf{x} \in \mathbb{R}^2$
- ▶ Konfigurační (kloubový) prostor: $\mathbf{q} \in \mathbb{R}^2$



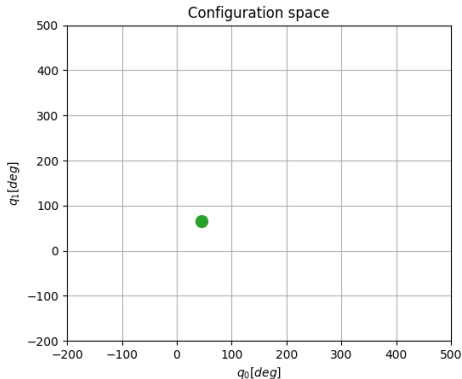
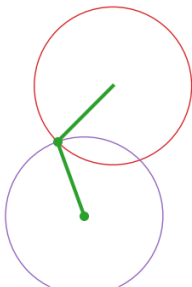
Příklad: Analytická IK pro strukturu RR

- ▶ Úkolový prostor: translace chapadla, $\boldsymbol{x} \in \mathbb{R}^2$
- ▶ Konfigurační (kloubový) prostor: $\boldsymbol{q} \in \mathbb{R}^2$



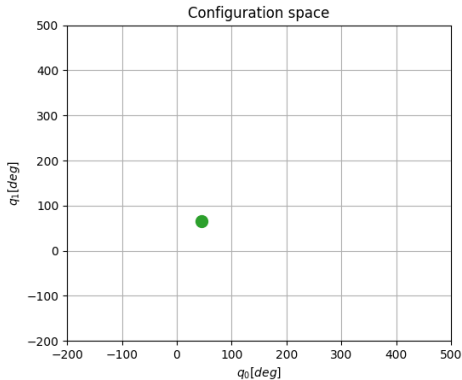
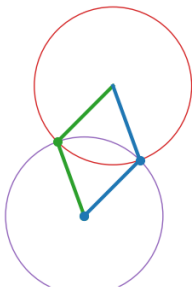
Příklad: Analytická IK pro strukturu RR

- ▶ Úkolový prostor: translace chapadla, $\mathbf{x} \in \mathbb{R}^2$
- ▶ Konfigurační (kloubový) prostor: $\mathbf{q} \in \mathbb{R}^2$



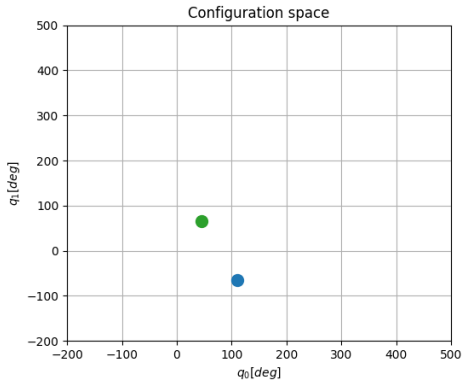
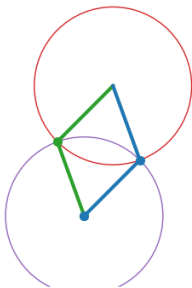
Příklad: Analytická IK pro strukturu RR

- ▶ Úkolový prostor: translace chapadla, $\boldsymbol{x} \in \mathbb{R}^2$
- ▶ Konfigurační (kloubový) prostor: $\boldsymbol{q} \in \mathbb{R}^2$



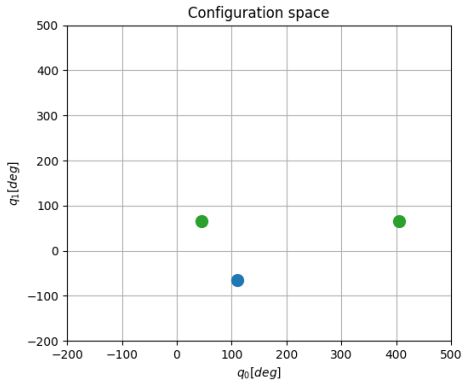
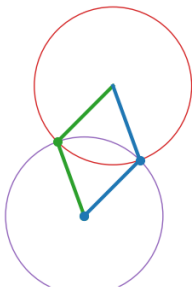
Příklad: Analytická IK pro strukturu RR

- ▶ Úkolový prostor: translace chapadla, $\mathbf{x} \in \mathbb{R}^2$
- ▶ Konfigurační (kloubový) prostor: $\mathbf{q} \in \mathbb{R}^2$



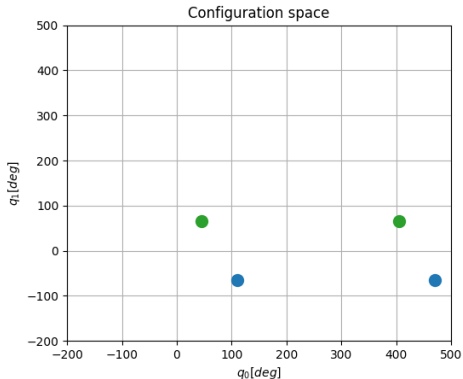
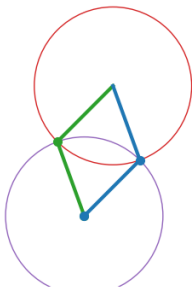
Příklad: Analytická IK pro strukturu RR

- ▶ Úkolový prostor: translace chapadla, $\mathbf{x} \in \mathbb{R}^2$
- ▶ Konfigurační (kloubový) prostor: $\mathbf{q} \in \mathbb{R}^2$



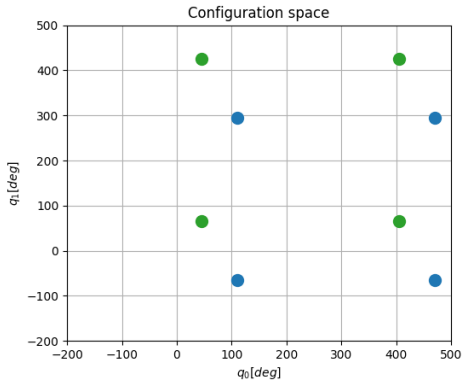
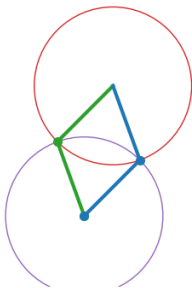
Příklad: Analytická IK pro strukturu RR

- ▶ Úkolový prostor: translace chapadla, $\mathbf{x} \in \mathbb{R}^2$
- ▶ Konfigurační (kloubový) prostor: $\mathbf{q} \in \mathbb{R}^2$



Příklad: Analytická IK pro strukturu RR

- ▶ Úkolový prostor: translace chapadla, $\mathbf{x} \in \mathbb{R}^2$
- ▶ Konfigurační (kloubový) prostor: $\mathbf{q} \in \mathbb{R}^2$



Příklad: Analytická IK pro strukturu RR

- ▶ Úkolový prostor: translace chapadla, $\mathbf{x} \in \mathbb{R}^2$
- ▶ Konfigurační (kloubový) prostor: $\mathbf{q} \in \mathbb{R}^2$
- ▶ Algoritmus:
 - ▶ Vypočítejte pozici (translaci) všech kloubů a chapadla
 - ▶ Žádné řešení, 1 řešení, 2 řešení, popř. ∞ řešení
 - ▶ Pro každé řešení vypočítejte konfigurace kloubů $\theta_i = \text{atan2}(y, x) + 2k\pi, k \in \mathbb{Z}$
 $\begin{pmatrix} x & y \end{pmatrix}^\top = \mathbf{t}_{i,i+1}$, t.j. translační část $T_{i,i+1}$



Numerická optimalizace

- ▶ Analytické řešení je často nedostupné
 - ▶ řešení neexistuje a hledáme nejbližší přibližné
 - ▶ existuje nekonečně mnoho řešení a my hledáme konfiguraci dle daného kritéria



Numerická optimalizace

- ▶ Analytické řešení je často nedostupné
 - ▶ řešení neexistuje a hledáme nejbližší přibližné
 - ▶ existuje nekonečně mnoho řešení a my hledáme konfiguraci dle daného kritéria
- ▶ Můžeme použít obecný numerický algoritmus, který iterativně snižuje chyby
- ▶ **Newton-Raphsonova** metoda



Numerická optimalizace

- ▶ Analytické řešení je často nedostupné
 - ▶ řešení neexistuje a hledáme nejbližší přibližné
 - ▶ existuje nekonečně mnoho řešení a my hledáme konfiguraci dle daného kritéria
- ▶ Můžeme použít obecný numerický algoritmus, který iterativně snižuje chyby
- ▶ **Newton-Raphsonova** metoda
 - ▶ řeší $g(\theta) = 0, g : \mathbb{R} \rightarrow \mathbb{R}$



Numerická optimalizace

- ▶ Analytické řešení je často nedostupné
 - ▶ řešení neexistuje a hledáme nejbližší přibližné
 - ▶ existuje nekonečně mnoho řešení a my hledáme konfiguraci dle daného kritéria
- ▶ Můžeme použít obecný numerický algoritmus, který iterativně snižuje chyby
- ▶ **Newton-Raphsonova** metoda
 - ▶ řeší $g(\theta) = 0, g : \mathbb{R} \rightarrow \mathbb{R}$
 - ▶ Taylorův rozvoj $g(\theta)$ v bodě θ^0 :
$$g(\theta) = g(\theta^0) + \frac{\partial g}{\partial \theta}(\theta^0)(\theta - \theta^0) + \text{členy vyššího řádu}$$



Numerická optimalizace

- ▶ Analytické řešení je často nedostupné
 - ▶ řešení neexistuje a hledáme nejbližší přibližné
 - ▶ existuje nekonečně mnoho řešení a my hledáme konfiguraci dle daného kritéria
- ▶ Můžeme použít obecný numerický algoritmus, který iterativně snižuje chyby
- ▶ **Newton-Raphsonova** metoda
 - ▶ řeší $g(\theta) = 0, g : \mathbb{R} \rightarrow \mathbb{R}$
 - ▶ Taylorův rozvoj $g(\theta)$ v bodě θ^0 :
$$g(\theta) = g(\theta^0) + \frac{\partial g}{\partial \theta}(\theta^0)(\theta - \theta^0) + \text{členy vyššího řádu}$$
 - ▶ dosazením $g(\theta) = 0$, ignorováním členů vyššího řádu a vyjmutím θ :
$$\theta \approx \theta^0 - \left(\frac{\partial g}{\partial \theta}(\theta^0) \right)^{-1} g(\theta^0)$$



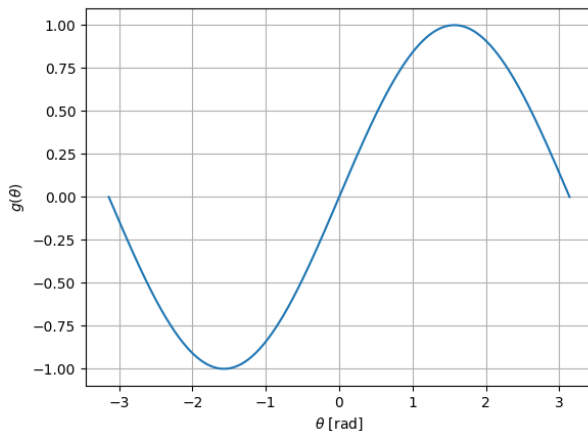
Numerická optimalizace

- ▶ Analytické řešení je často nedostupné
 - ▶ řešení neexistuje a hledáme nejbližší přibližné
 - ▶ existuje nekonečně mnoho řešení a my hledáme konfiguraci dle daného kritéria
- ▶ Můžeme použít obecný numerický algoritmus, který iterativně snižuje chyby
- ▶ **Newton-Raphsonova** metoda
 - ▶ řeší $g(\theta) = 0, g : \mathbb{R} \rightarrow \mathbb{R}$
 - ▶ Taylorův rozvoj $g(\theta)$ v bodě θ^0 :
$$g(\theta) = g(\theta^0) + \frac{\partial g}{\partial \theta}(\theta^0)(\theta - \theta^0) + \text{členy vyššího řádu}$$
 - ▶ dosazením $g(\theta) = 0$, ignorováním členů vyššího řádu a vyjmutím θ :
$$\theta \approx \theta^0 - \left(\frac{\partial g}{\partial \theta}(\theta^0) \right)^{-1} g(\theta^0)$$
 - ▶ protože ignorujeme členy vyššího řádu, musíme iterovat:
$$\theta^{k+1} = \theta^k - \left(\frac{\partial g}{\partial \theta}(\theta^k) \right)^{-1} g(\theta^k)$$



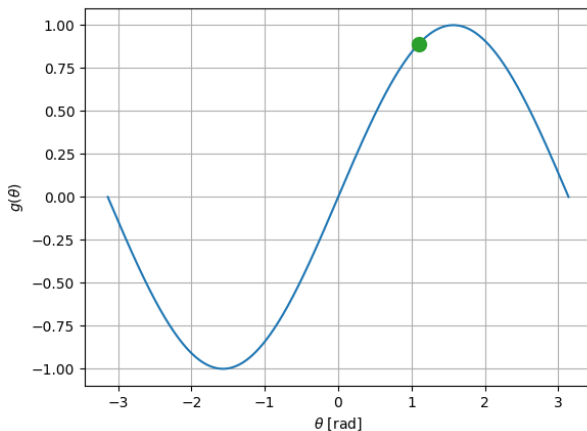
Příklad 1D Newton-Raphsonovy metody

- $g(\theta) = \sin(\theta)$, nalezni θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1, 1$



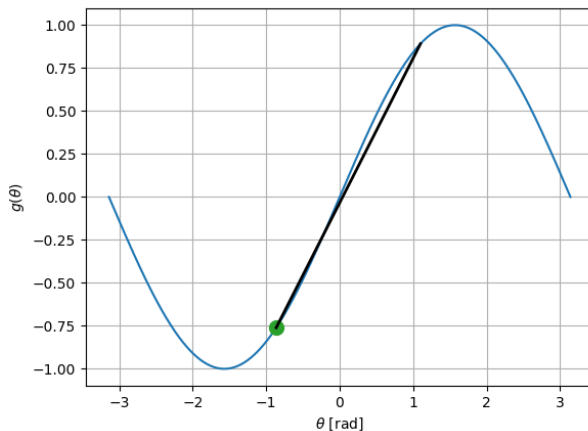
Příklad 1D Newton-Raphsonovy metody

- $g(\theta) = \sin(\theta)$, nalezni θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,1$



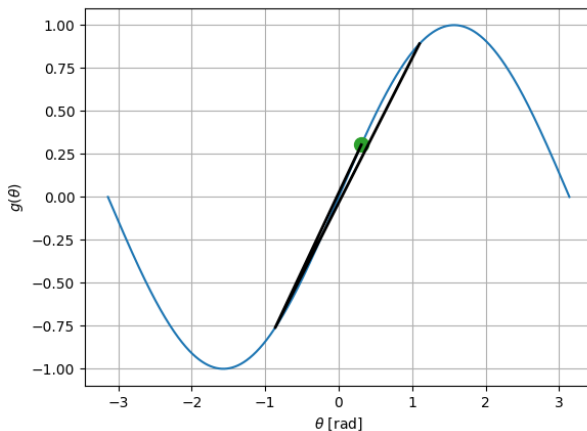
Příklad 1D Newton-Raphsonovy metody

- $g(\theta) = \sin(\theta)$, nalezni θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,1$



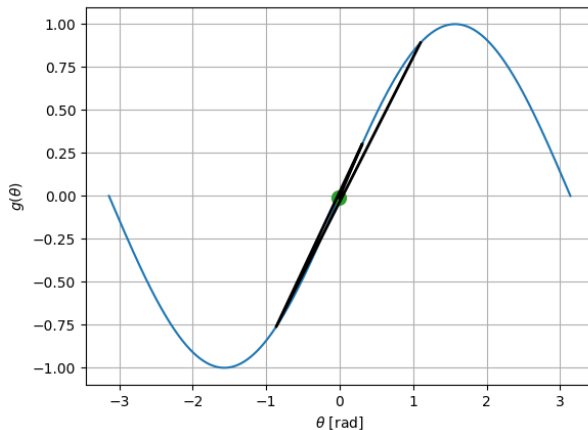
Příklad 1D Newton-Raphsonovy metody

- $g(\theta) = \sin(\theta)$, nalezni θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,1$



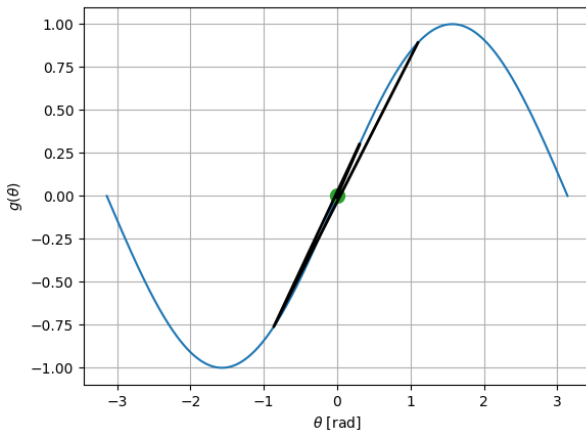
Příklad 1D Newton-Raphsonovy metody

- $g(\theta) = \sin(\theta)$, nalezni θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,1$



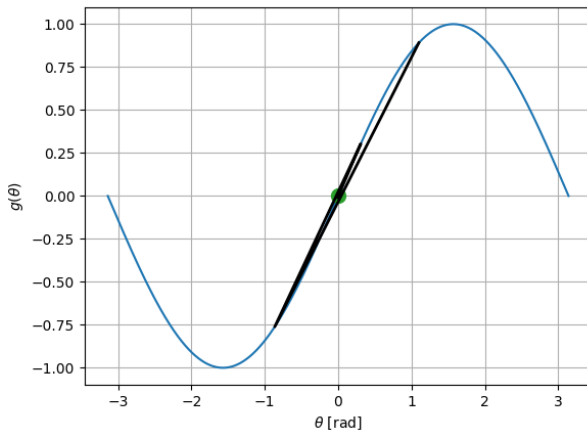
Příklad 1D Newton-Raphsonovy metody

- $g(\theta) = \sin(\theta)$, nalezni θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,1$



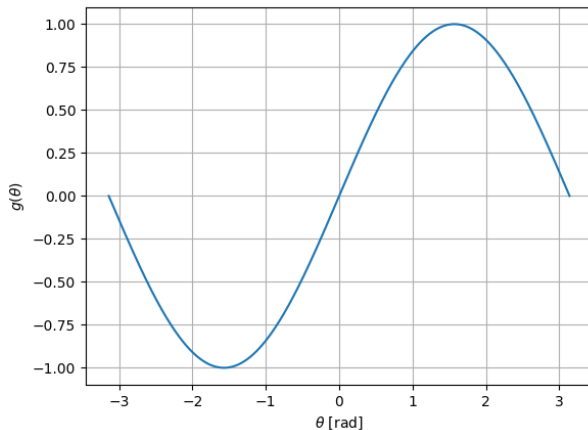
Příklad 1D Newton-Raphsonovy metody

- $g(\theta) = \sin(\theta)$, najezni θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,1$



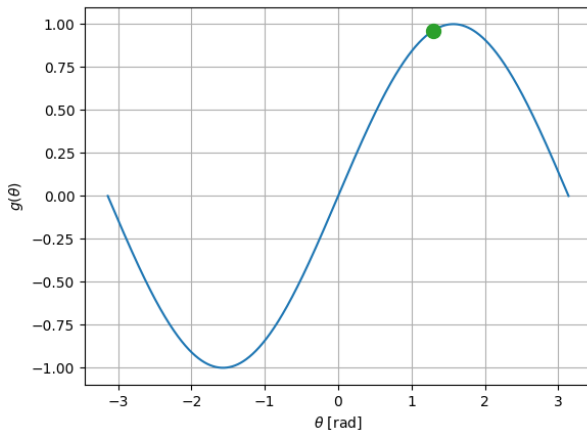
Příklad 1D Newton-Raphsonovy metody

- ▶ $g(\theta) = \sin(\theta)$, najdi θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,3$
- ▶ Kvalita řešení závisí na počátečním odhadu



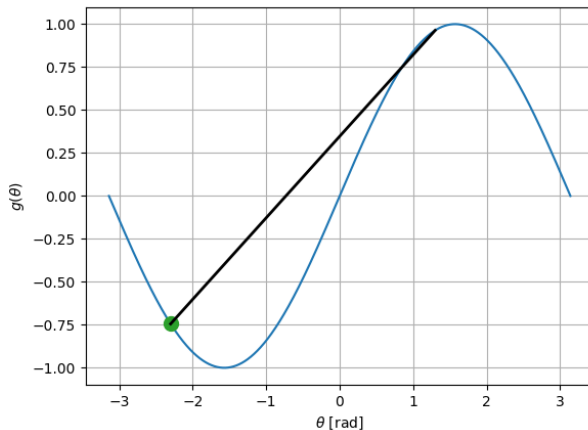
Příklad 1D Newton-Raphsonovy metody

- ▶ $g(\theta) = \sin(\theta)$, najdi θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,3$
- ▶ Kvalita řešení závisí na počátečním odhadu



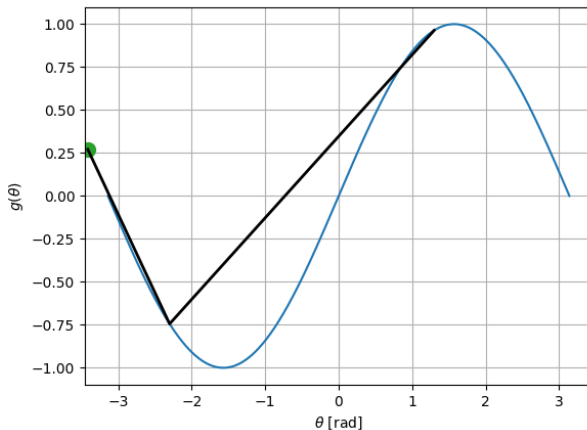
Příklad 1D Newton-Raphsonovy metody

- ▶ $g(\theta) = \sin(\theta)$, najdi θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,3$
- ▶ Kvalita řešení závisí na počátečním odhadu



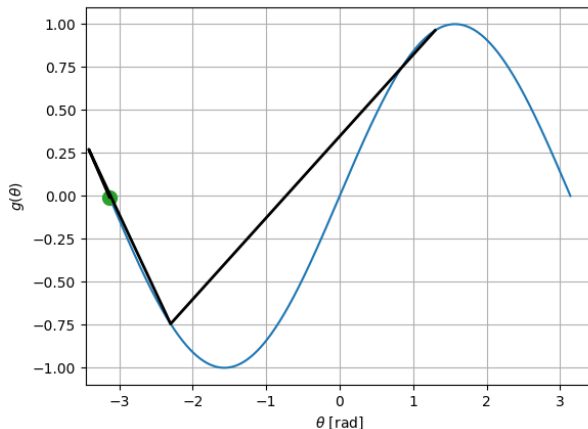
Příklad 1D Newton-Raphsonovy metody

- ▶ $g(\theta) = \sin(\theta)$, najdi θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,3$
- ▶ Kvalita řešení závisí na počátečním odhadu



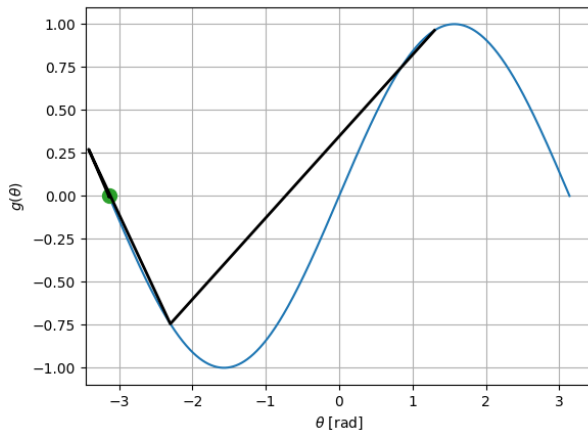
Příklad 1D Newton-Raphsonovy metody

- ▶ $g(\theta) = \sin(\theta)$, najdi θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,3$
- ▶ Kvalita řešení závisí na počátečním odhadu



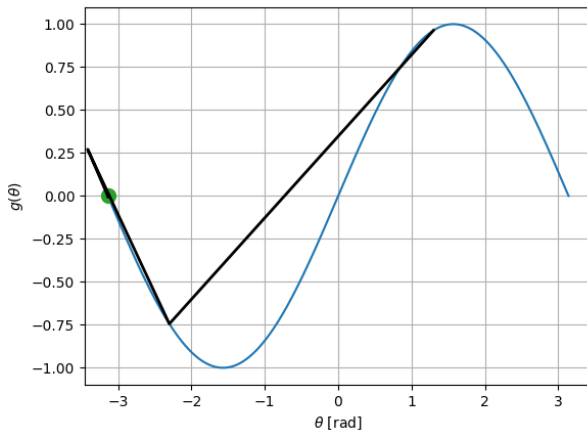
Příklad 1D Newton-Raphsonovy metody

- ▶ $g(\theta) = \sin(\theta)$, najdi θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,3$
- ▶ Kvalita řešení závisí na počátečním odhadu



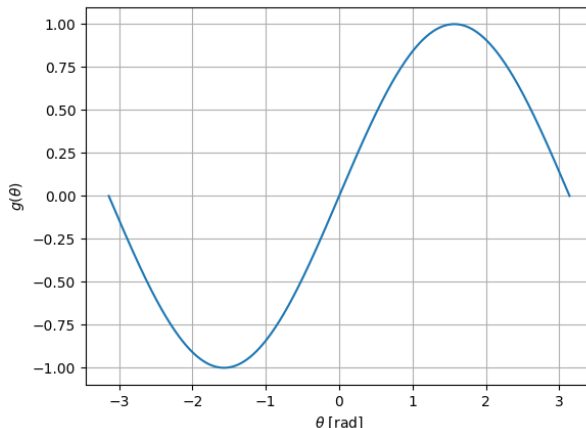
Příklad 1D Newton-Raphsonovy metody

- ▶ $g(\theta) = \sin(\theta)$, najdi θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,3$
- ▶ Kvalita řešení závisí na počátečním odhadu



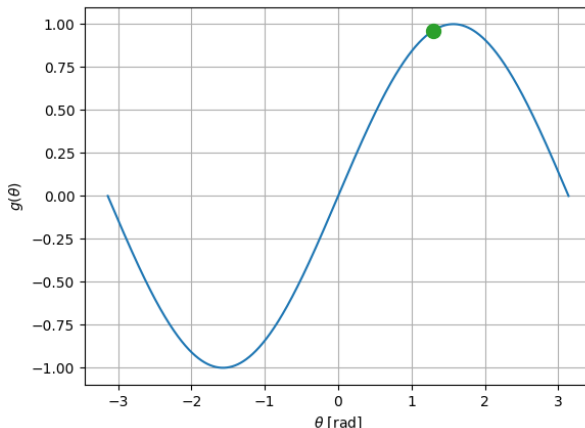
Příklad 1D Newton-Raphsonovy metody

- ▶ $g(\theta) = \sin(\theta)$, najdi θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,3$, $\alpha = 0,5$
- ▶ $\theta^{k+1} = \theta^k - \alpha \left(\frac{\partial g}{\partial \theta}(\theta^k) \right)^{-1} g(\theta^k)$



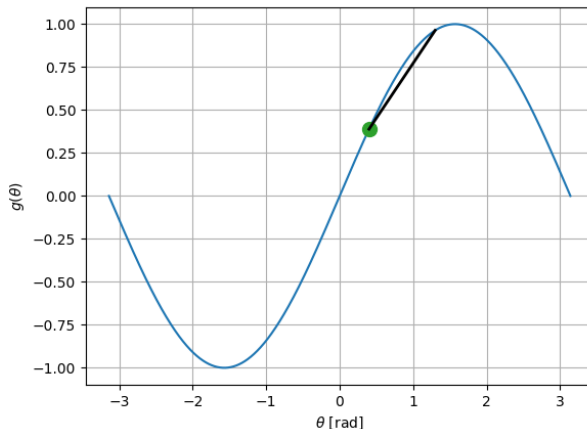
Příklad 1D Newton-Raphsonovy metody

- ▶ $g(\theta) = \sin(\theta)$, najdi θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,3$, $\alpha = 0,5$
- ▶ $\theta^{k+1} = \theta^k - \alpha \left(\frac{\partial g}{\partial \theta}(\theta^k) \right)^{-1} g(\theta^k)$



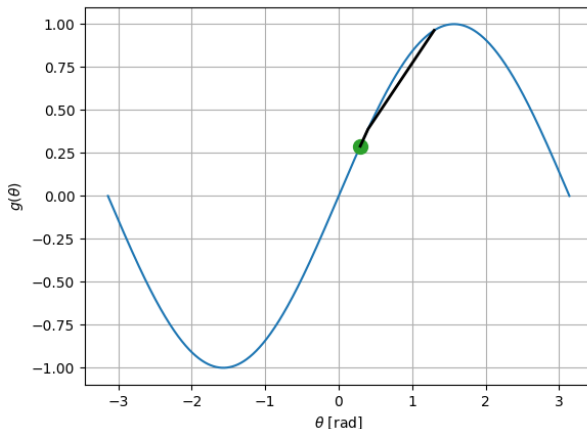
Příklad 1D Newton-Raphsonovy metody

- ▶ $g(\theta) = \sin(\theta)$, najdi θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,3$, $\alpha = 0,5$
- ▶ $\theta^{k+1} = \theta^k - \alpha \left(\frac{\partial g}{\partial \theta}(\theta^k) \right)^{-1} g(\theta^k)$



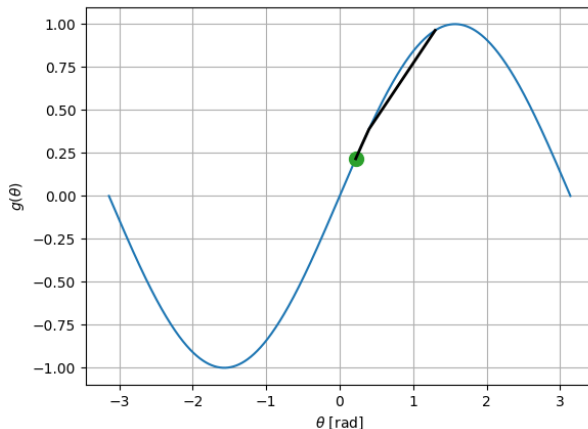
Příklad 1D Newton-Raphsonovy metody

- ▶ $g(\theta) = \sin(\theta)$, najdi θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1, 3$, $\alpha = 0, 5$
- ▶ $\theta^{k+1} = \theta^k - \alpha \left(\frac{\partial g}{\partial \theta}(\theta^k) \right)^{-1} g(\theta^k)$



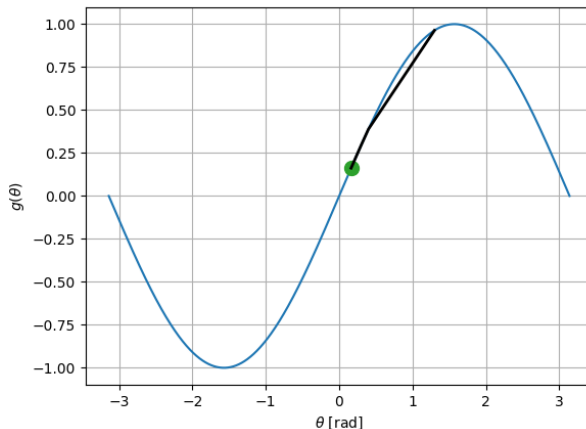
Příklad 1D Newton-Raphsonovy metody

- ▶ $g(\theta) = \sin(\theta)$, najdi θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,3$, $\alpha = 0,5$
- ▶ $\theta^{k+1} = \theta^k - \alpha \left(\frac{\partial g}{\partial \theta}(\theta^k) \right)^{-1} g(\theta^k)$



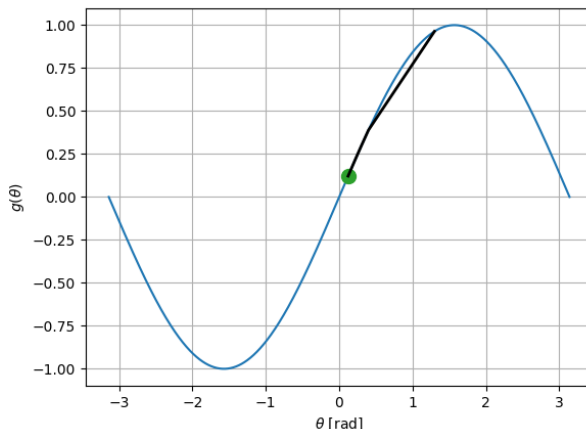
Příklad 1D Newton-Raphsonovy metody

- ▶ $g(\theta) = \sin(\theta)$, najdi θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,3$, $\alpha = 0,5$
- ▶ $\theta^{k+1} = \theta^k - \alpha \left(\frac{\partial g}{\partial \theta}(\theta^k) \right)^{-1} g(\theta^k)$



Příklad 1D Newton-Raphsonovy metody

- ▶ $g(\theta) = \sin(\theta)$, najdi θ^* tak, aby $g(\theta^*) = 0$, $\theta^0 = 1,3$, $\alpha = 0,5$
- ▶ $\theta^{k+1} = \theta^k - \alpha \left(\frac{\partial g}{\partial \theta}(\theta^k) \right)^{-1} g(\theta^k)$



Jak najít α ?

- ▶ Line-search algoritmus



Jak najít α ?

- ▶ Line-search algoritmus
- ▶ Nalezni α tak, aby $g(\theta^{k+1}) < g(\theta^k)$



Jak najít α ?

- ▶ Line-search algoritmus
- ▶ Nalezni α tak, aby $g(\theta^{k+1}) < g(\theta^k)$
- ▶ Algoritmus:
 - ▶ $\alpha^0 = 1$



Jak najít α ?

- ▶ Line-search algoritmus
- ▶ Nalezni α tak, aby $g(\theta^{k+1}) < g(\theta^k)$
- ▶ Algoritmus:
 - ▶ $\alpha^0 = 1$
 - ▶ jestli $g(\theta^{k+1}) < g(\theta^k)$: konec



Jak najít α ?

- ▶ Line-search algoritmus
- ▶ Nalezni α tak, aby $g(\theta^{k+1}) < g(\theta^k)$
- ▶ Algoritmus:
 - ▶ $\alpha^0 = 1$
 - ▶ jestli $g(\theta^{k+1}) < g(\theta^k)$: konec
 - ▶ $\alpha^{i+1} = \tau \alpha^i$, $0 < \tau < 1$, např. $\tau = 0,5$
 - ▶ opakuj



Jak najít α ?

- ▶ Line-search algoritmus
- ▶ Nalezni α tak, aby $g(\theta^{k+1}) < g(\theta^k)$
- ▶ Algoritmus:
 - ▶ $\alpha^0 = 1$
 - ▶ jestli $g(\theta^{k+1}) < g(\theta^k)$: konec
 - ▶ $\alpha^{i+1} = \tau \alpha^i$, $0 < \tau < 1$, např. $\tau = 0,5$
 - ▶ opakuj
- ▶ Existují sofistikovanější algoritmy pro line-search

Numerické řešení pro RR IK

- ▶ Newton–Raphsonova metoda pro n -rozměrný případ

$$\boldsymbol{\theta}^{k+1} = \boldsymbol{\theta}^k - \alpha \left(\frac{\partial g}{\partial \boldsymbol{\theta}}(\boldsymbol{\theta}^k) \right)^{-1} g(\boldsymbol{\theta}^k) \text{ řeší } g(\boldsymbol{\theta}) = \mathbf{0}$$



Numerické řešení pro RR IK

- ▶ Newton–Raphsonova metoda pro n -rozměrný případ

$$\boldsymbol{\theta}^{k+1} = \boldsymbol{\theta}^k - \alpha \left(\frac{\partial g}{\partial \boldsymbol{\theta}}(\boldsymbol{\theta}^k) \right)^{-1} g(\boldsymbol{\theta}^k) \text{ řeší } g(\boldsymbol{\theta}) = \mathbf{0}$$

- ▶ Pro kinematiku manipulátoru:

$$g(\mathbf{q}) = \mathbf{x}_d - f_{\text{fk}}(\mathbf{q}), \mathbf{x}_d \in \mathbb{R}^2$$



Numerické řešení pro RR IK

- ▶ Newton–Raphsonova metoda pro n -rozměrný případ

$$\boldsymbol{\theta}^{k+1} = \boldsymbol{\theta}^k - \alpha \left(\frac{\partial g}{\partial \boldsymbol{\theta}}(\boldsymbol{\theta}^k) \right)^{-1} g(\boldsymbol{\theta}^k) \text{ řeší } g(\boldsymbol{\theta}) = \mathbf{0}$$

- ▶ Pro kinematiku manipulátoru:

$$g(\mathbf{q}) = \mathbf{x}_d - f_{\text{fk}}(\mathbf{q}), \mathbf{x}_d \in \mathbb{R}^2$$

- ▶ Podle metody NR (pro $g(\mathbf{q}) = 0$):

$$\mathbf{x}_d = f_{\text{fk}}(\mathbf{q}_d) \approx f_{\text{fk}}(\mathbf{q}^0) + \frac{\partial f_{\text{fk}}}{\partial \mathbf{q}}(\mathbf{q}^0)(\mathbf{q}_d - \mathbf{q}^0)$$



Numerické řešení pro RR IK

- ▶ Newton–Raphsonova metoda pro n -rozměrný případ

$$\boldsymbol{\theta}^{k+1} = \boldsymbol{\theta}^k - \alpha \left(\frac{\partial g}{\partial \boldsymbol{\theta}}(\boldsymbol{\theta}^k) \right)^{-1} g(\boldsymbol{\theta}^k) \text{ řeší } g(\boldsymbol{\theta}) = \mathbf{0}$$

- ▶ Pro kinematiku manipulátoru:

$$g(\mathbf{q}) = \mathbf{x}_d - f_{\text{fk}}(\mathbf{q}), \mathbf{x}_d \in \mathbb{R}^2$$

- ▶ Podle metody NR (pro $g(\mathbf{q}) = 0$):

$$\mathbf{x}_d = f_{\text{fk}}(\mathbf{q}_d) \approx f_{\text{fk}}(\mathbf{q}^0) + \frac{\partial f_{\text{fk}}}{\partial \mathbf{q}}(\mathbf{q}^0)(\mathbf{q}_d - \mathbf{q}^0) = f_{\text{fk}}(\mathbf{q}^0) + J(\mathbf{q}^0)(\mathbf{q}_d - \mathbf{q}^0)$$



Numerické řešení pro RR IK

- ▶ Newton–Raphsonova metoda pro n -rozměrný případ

$$\boldsymbol{\theta}^{k+1} = \boldsymbol{\theta}^k - \alpha \left(\frac{\partial g}{\partial \boldsymbol{\theta}}(\boldsymbol{\theta}^k) \right)^{-1} g(\boldsymbol{\theta}^k) \text{ řeší } g(\boldsymbol{\theta}) = \mathbf{0}$$

- ▶ Pro kinematiku manipulátoru:

$$g(\mathbf{q}) = \mathbf{x}_d - f_{\text{fk}}(\mathbf{q}), \mathbf{x}_d \in \mathbb{R}^2$$

- ▶ Podle metody NR (pro $g(\mathbf{q}) = 0$):

$$\begin{aligned} \mathbf{x}_d &= f_{\text{fk}}(\mathbf{q}_d) \approx f_{\text{fk}}(\mathbf{q}^0) + \frac{\partial f_{\text{fk}}}{\partial \mathbf{q}}(\mathbf{q}^0)(\mathbf{q}_d - \mathbf{q}^0) = f_{\text{fk}}(\mathbf{q}^0) + J(\mathbf{q}^0)(\mathbf{q}_d - \mathbf{q}^0) \\ \mathbf{q}_d &\approx \mathbf{q}^0 + J(\mathbf{q}^0)^{-1}(\mathbf{x}_d - f_{\text{fk}}(\mathbf{q}^0)) \end{aligned}$$



Numerické řešení pro RR IK

- ▶ Newton–Raphsonova metoda pro n -rozměrný případ

$$\boldsymbol{\theta}^{k+1} = \boldsymbol{\theta}^k - \alpha \left(\frac{\partial g}{\partial \boldsymbol{\theta}}(\boldsymbol{\theta}^k) \right)^{-1} g(\boldsymbol{\theta}^k) \text{ řeší } g(\boldsymbol{\theta}) = 0$$

- ▶ Pro kinematiku manipulátoru:

$$g(\mathbf{q}) = \mathbf{x}_d - f_{fk}(\mathbf{q}), \mathbf{x}_d \in \mathbb{R}^2$$

- ▶ Podle metody NR (pro $g(\mathbf{q}) = 0$):

$$\begin{aligned} \mathbf{x}_d = f_{fk}(\mathbf{q}_d) &\approx f_{fk}(\mathbf{q}^0) + \frac{\partial f_{fk}}{\partial \mathbf{q}}(\mathbf{q}^0)(\mathbf{q}_d - \mathbf{q}^0) = f_{fk}(\mathbf{q}^0) + J(\mathbf{q}^0)(\mathbf{q}_d - \mathbf{q}^0) \\ \mathbf{q}_d &\approx \mathbf{q}^0 + J(\mathbf{q}^0)^{-1}(\mathbf{x}_d - f_{fk}(\mathbf{q}^0)) \end{aligned}$$

- ▶ Iterativně pomocí line-search:

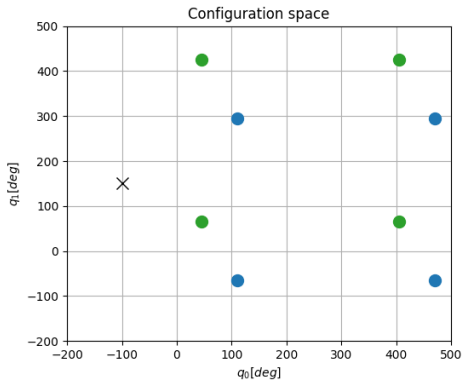
$$\mathbf{q}^{k+1} = \mathbf{q}^k + \alpha J(\mathbf{q}^k)^{-1}(\mathbf{x}_d - f_{fk}(\mathbf{q}^k))$$

- ▶ Intuice skrze diferenciální kinematiku:

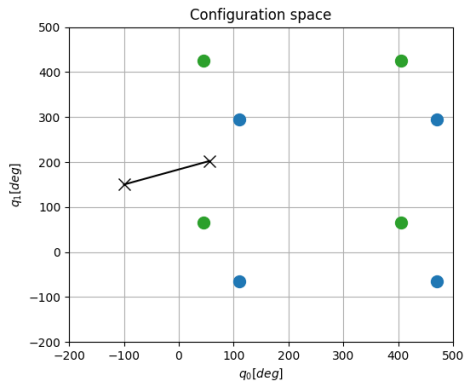
- ▶ jaká by měla být rychlost v kloubovém prostoru tak, abychom dosáhli dané rychlosti v úkolovém prostoru
- ▶ $\dot{\mathbf{q}} = J^{-1}\dot{\mathbf{x}}$



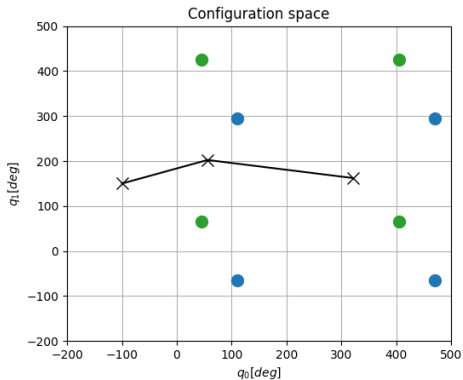
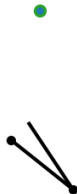
Numerické řešení pro RR IK #1



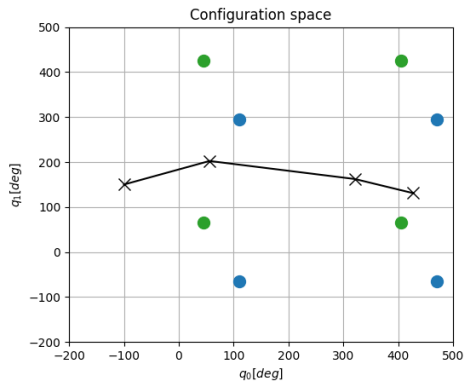
Numerické řešení pro RR IK #1



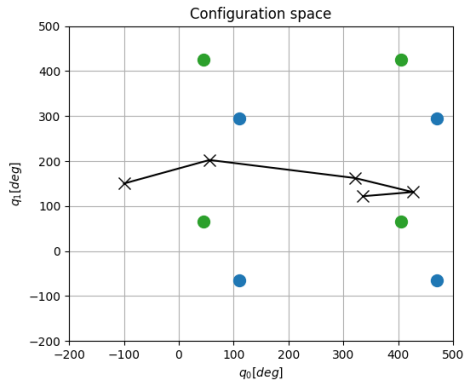
Numerické řešení pro RR IK #1



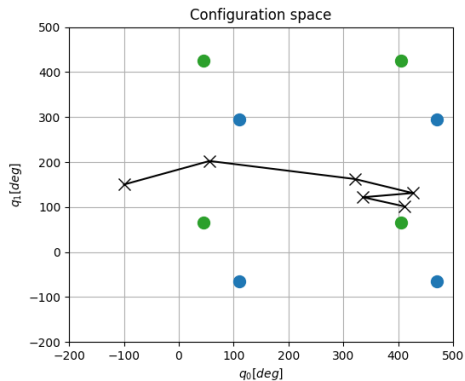
Numerické řešení pro RR IK #1



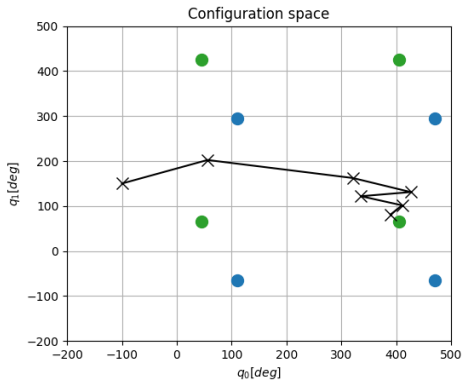
Numerické řešení pro RR IK #1



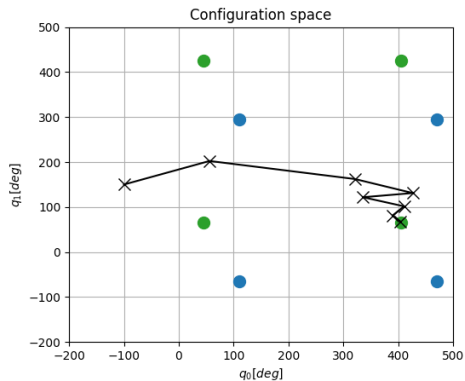
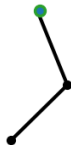
Numerické řešení pro RR IK #1



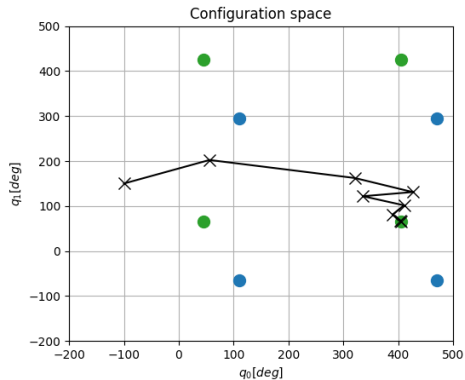
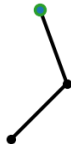
Numerické řešení pro RR IK #1



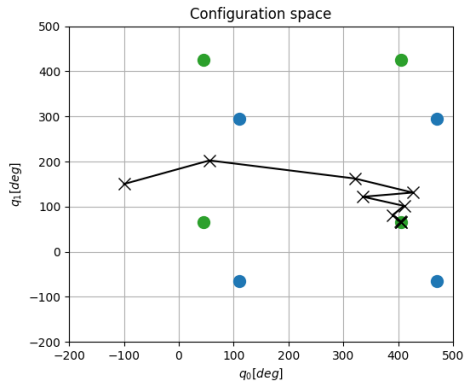
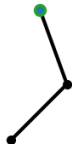
Numerické řešení pro RR IK #1



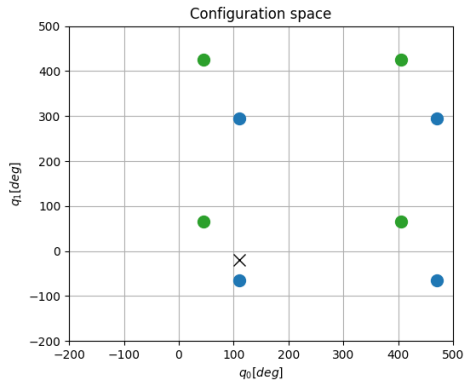
Numerické řešení pro RR IK #1



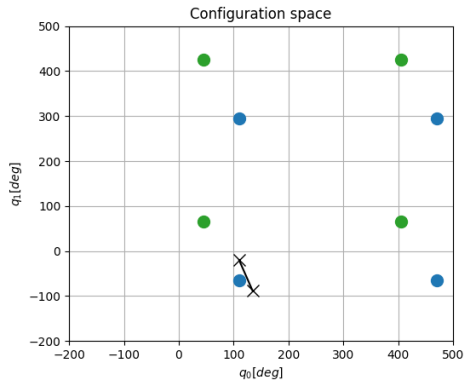
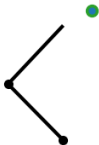
Numerické řešení pro RR IK #1



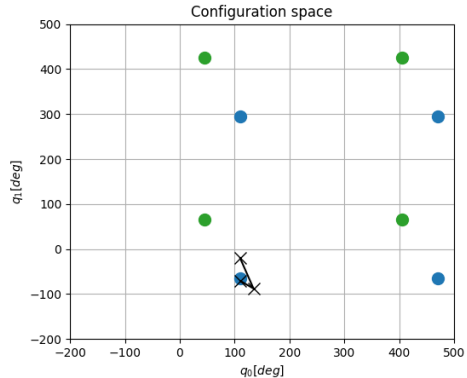
Numerické řešení pro RR IK #2



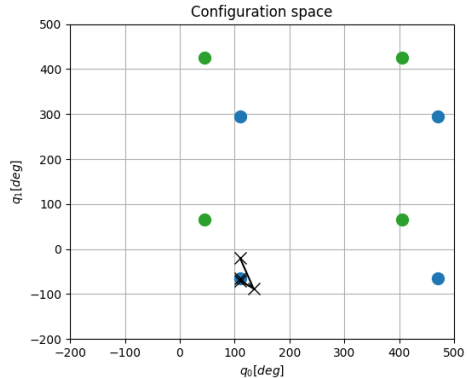
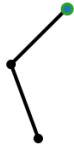
Numerické řešení pro RR IK #2



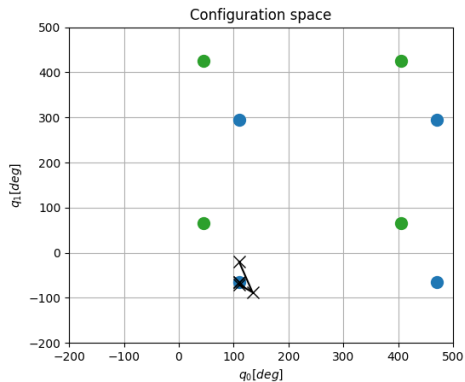
Numerické řešení pro RR IK #2



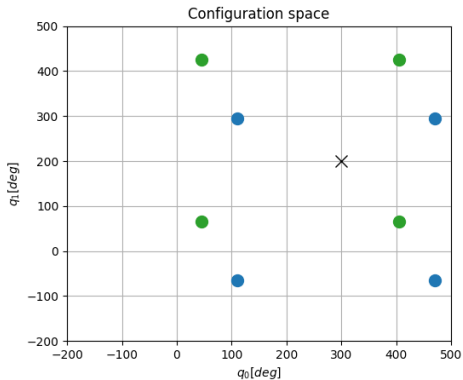
Numerické řešení pro RR IK #2



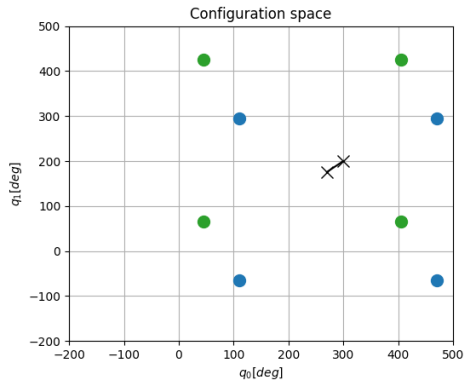
Numerické řešení pro RR IK #2



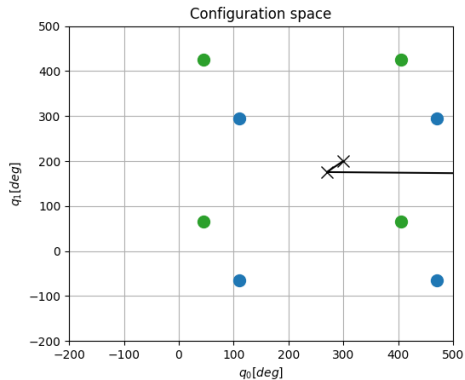
Numerické řešení pro RR IK #3



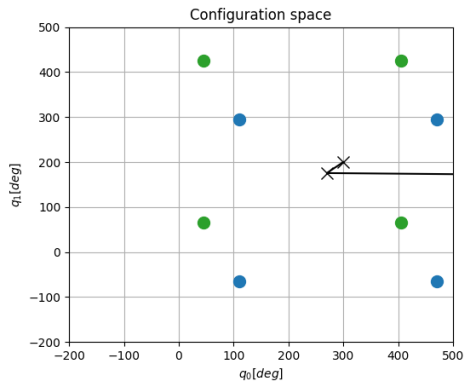
Numerické řešení pro RR IK #3



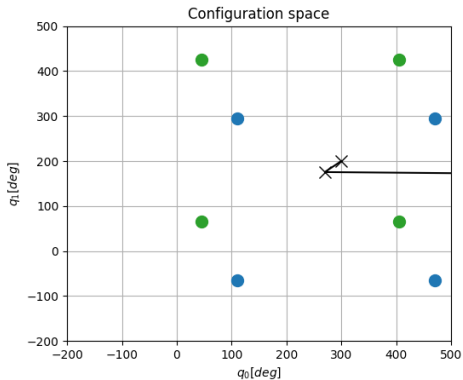
Numerické řešení pro RR IK #3



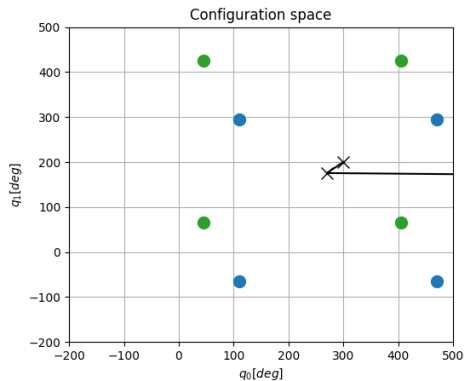
Numerické řešení pro RR IK #3



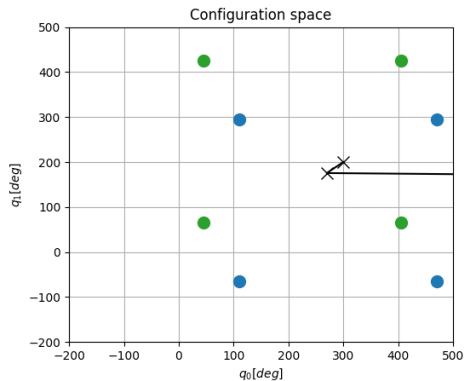
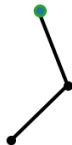
Numerické řešení pro RR IK #3



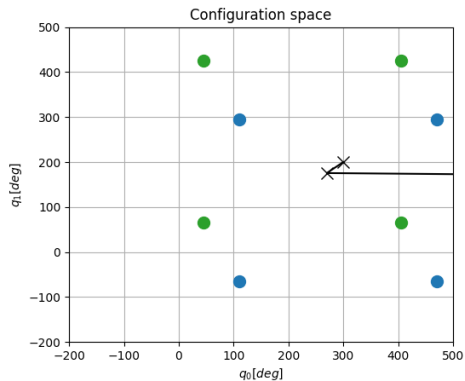
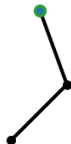
Numerické řešení pro RR IK #3



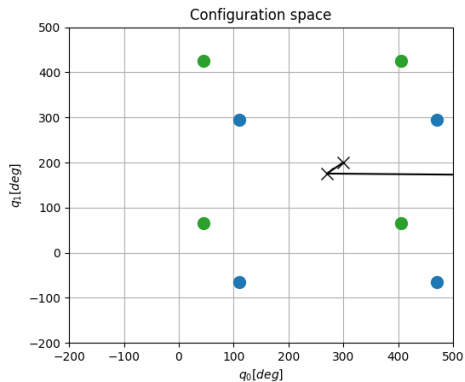
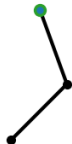
Numerické řešení pro RR IK #3



Numerické řešení pro RR IK #3



Numerické řešení pro RR IK #3



Numerické řešení - shrnutí

- ▶ Numerické řešení je snadno implementovatelné pro obecné manipulátory
- ▶ Prvotní odhad je důležitý
 - ▶ pokud jsme *blízko* řešení, FK je téměř lineární, budeme konvergovat k *nejbližšímu* řešení
 - ▶ pokud jsme příliš *daleko*, nemáme žádnou kontrolu nad tím, jaké řešení je vybráno
 - ▶ může pomoci vyladění velikosti kroku
- ▶ Musíme definovat kritérium konvergence
 - ▶ např. $\|x_d - f_{fk}(q^k)\| < \varepsilon$



Co když J není invertibilní?

- ▶ Redundantní roboti, podaktuovaní roboti, singularita



Co když J není invertibilní?

- ▶ Redundantní roboti, podaktuovaní roboti, singularita
- ▶ Moore-Penroseova pseudoinverze $J^\dagger$
- ▶ $\mathbf{q}^{k+1} = \mathbf{q}^k + \alpha J^\dagger(\mathbf{q}^k)(\mathbf{x}_d - f_{fk}(\mathbf{q}^k))$



Co když J není invertibilní?

- ▶ Redundantní roboti, podaktuovaní roboti, singularita
- ▶ Moore-Penroseova pseudoinverze $J^\dagger$
- ▶ $\mathbf{q}^{k+1} = \mathbf{q}^k + \alpha J^\dagger(\mathbf{q}^k)(\mathbf{x}_d - f_{fk}(\mathbf{q}^k))$
- ▶ Redundantní roboti
 - ▶ nekonečno řešení pro dosažení stejné rychlosti v prostoru úkolu
 - ▶ pseudoinverze navíc minimalizuje $\|\mathbf{q}\|$

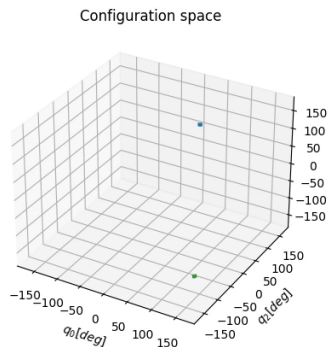


Co když J není invertibilní?

- ▶ Redundantní roboti, podaktuovaní roboti, singularita
- ▶ Moore-Penroseova pseudoinverze $J^\dagger$
- ▶ $\mathbf{q}^{k+1} = \mathbf{q}^k + \alpha J^\dagger(\mathbf{q}^k)(\mathbf{x}_d - f_{fk}(\mathbf{q}^k))$
- ▶ Redundantní roboti
 - ▶ nekonečno řešení pro dosažení stejné rychlosti v prostoru úkolu
 - ▶ pseudoinverze navíc minimalizuje $\|\mathbf{q}\|$
- ▶ Podaktuovaní roboti nebo singularita
 - ▶ neexistuje žádné přesné řešení pro rychlost úkolového prostoru
 - ▶ pseudoinverze minimalizuje chybu v prostoru úloh



IK řešení pro redundantního robota



IK v $SE(2)$ pro RRR

- ▶ Daná požadovaná transformace $T_{RG}^D \in SE(2)$
 - ▶ R - referenční s.s.
 - ▶ G - s.s. chapadla (gripper)



IK v $SE(2)$ pro RRR

- ▶ Daná požadovaná transformace $T_{RG}^D \in SE(2)$
 - ▶ R - referenční s.s.
 - ▶ G - s.s. chapadla (gripper)
- ▶ Analytické řešení
 - ▶ rozdělit problém na rotaci (poslední kloub) a polohu (ostatní klouby)
 - ▶ $t_{RC} = T_{RG}^D (-l_3 \ 0 \ 1)^\top$
 - ▶ t_{RB} počítejte jako pro RR pro translační úkolový prostor
 - ▶ použijte atan2 pro výpočet konfigurace kloubů



IK v $SE(2)$ pro RRR

- ▶ Daná požadovaná transformace $T_{RG}^D \in SE(2)$
 - ▶ R - referenční s.s.
 - ▶ G - s.s. chapadla (gripper)
- ▶ Analytické řešení
 - ▶ rozdělit problém na rotaci (poslední kloub) a polohu (ostatní klouby)
 - ▶ $t_{RC} = T_{RG}^D (-l_3 \ 0 \ 1)^\top$
 - ▶ t_{RB} počítejte jako pro RR pro translační úkolový prostor
 - ▶ použijte atan2 pro výpočet konfigurace kloubů
- ▶ Numerické řešení
 - ▶ chyba v referenčním s.s:
$$e(\mathbf{q}) = (x_{RG}^D - x_{RG}(\mathbf{q}) \quad y_{RG}^D - y_{RG}(\mathbf{q}) \quad \phi_{RG}^D - \phi_{RG}(\mathbf{q}))^\top$$



IK v $SE(2)$ pro RRR

- ▶ Daná požadovaná transformace $T_{RG}^D \in SE(2)$
 - ▶ R - referenční s.s.
 - ▶ G - s.s. chapadla (gripper)
- ▶ Analytické řešení
 - ▶ rozdělit problém na rotaci (poslední kloub) a polohu (ostatní klouby)
 - ▶ $t_{RC} = T_{RG}^D \begin{pmatrix} -l_3 & 0 & 1 \end{pmatrix}^\top$
 - ▶ t_{RB} počítejte jako pro RR pro translační úkolový prostor
 - ▶ použijte atan2 pro výpočet konfigurace kloubů
- ▶ Numerické řešení
 - ▶ chyba v referenčním s.s:
$$e(\mathbf{q}) = \begin{pmatrix} x_{RG}^D - x_{RG}(\mathbf{q}) & y_{RG}^D - y_{RG}(\mathbf{q}) & \phi_{RG}^D - \phi_{RG}(\mathbf{q}) \end{pmatrix}^\top$$
 - ▶ NR iterace:
$$\mathbf{q}^{k+1} = \mathbf{q}^k + \alpha J^\dagger(\mathbf{q}^k) e(\mathbf{q}^k)$$

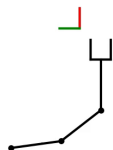


IK v $SE(2)$ pro RRR

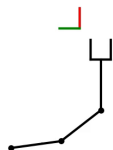
- ▶ Daná požadovaná transformace $T_{RG}^D \in SE(2)$
 - ▶ R - referenční s.s.
 - ▶ G - s.s. chapadla (gripper)
- ▶ Analytické řešení
 - ▶ rozdělit problém na rotaci (poslední kloub) a polohu (ostatní klouby)
 - ▶ $t_{RC} = T_{RG}^D (-l_3 \ 0 \ 1)^\top$
 - ▶ t_{RB} počítejte jako pro RR pro translační úkolový prostor
 - ▶ použijte atan2 pro výpočet konfigurace kloubů
- ▶ Numerické řešení
 - ▶ chyba v referenčním s.s:
$$e(\mathbf{q}) = (x_{RG}^D - x_{RG}(\mathbf{q}) \quad y_{RG}^D - y_{RG}(\mathbf{q}) \quad \phi_{RG}^D - \phi_{RG}(\mathbf{q}))^\top$$
 - ▶ NR iterace:
$$\mathbf{q}^{k+1} = \mathbf{q}^k + \alpha J^\dagger(\mathbf{q}^k) e(\mathbf{q}^k)$$



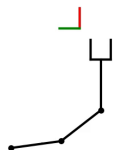
Numerické řešení v $SE(2)$



Numerické řešení v $SE(2)$



Numerické řešení v $SE(2)$



IK v $SE(3)$

- ▶ Numerický IK algoritmus je téměř stejný
 - ▶ chyba se musí vypočítat pomocí transformací
 - ▶ stejně jako v rovinném případě, chyba musí být reprezentována v referenčním s.s.



IK v $SE(3)$

- ▶ Numerický IK algoritmus je téměř stejný
 - ▶ chyba se musí vypočítat pomocí transformací
 - ▶ stejně jako v rovinném případě, chyba musí být reprezentována v referenčním s.s.
- ▶ Analytické řešení nemusí pro obecný 6 DoF manipulátor existovat



- ▶ Numerický IK algoritmus je téměř stejný
 - ▶ chyba se musí vypočítat pomocí transformací
 - ▶ stejně jako v rovinném případě, chyba musí být reprezentována v referenčním s.s.
- ▶ Analytické řešení nemusí pro obecný 6 DoF manipulátor existovat
- ▶ Pro prostorového 6 DoF robota s otočnými klouby
 - ▶ řešení lze oddělit, pokud se poslední tři kloubové osy vzájemně protínají
 - ▶ poslední tři klouby se použijí k orientaci chapadla
 - ▶ první tři klouby se použijí k polohování příruby



Příklad důležitosti více řešení



- ▶ Inverzní kinematika
 - ▶ analytické řešení pomocí geometrické analýzy
 - ▶ vede k výpočtu průniků geometrických primitiv
 - ▶ numerické řešení, Newton–Raphsonova metoda
 - ▶ Jakobián
 - ▶ pseudoinverze
- ▶ Počet řešení inverzní kinematiky
 - ▶ žádné řešení
 - ▶ více řešení
 - ▶ periodické řešení
 - ▶ nekonečný počet řešení

- ▶ Numerická IK v $SE(2)$
- ▶ Analytická IK v $SE(2)$ pro RRR manipulátor
- ▶ Analytická IK v $SE(2)$ pro PRR manipulátor [dobrovolné]