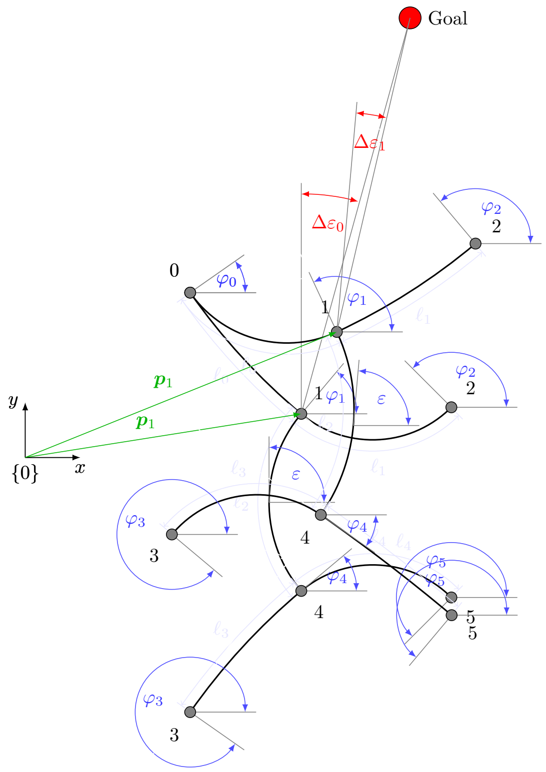
1 Problem Statement

- Angenommen die Konfiguration / Pose des Roboters $\boldsymbol{\rho} = [\boldsymbol{\alpha}, \boldsymbol{p}_1, \varepsilon]$ ist vollständig bekannt, wobei $\boldsymbol{\alpha}$ die Gelenkkoordinaten / Biegewinkel der einzelnen Glieder sind, $\boldsymbol{p}_1$ die Position des vorderen Torsoendes und ε die Orientierung des Roboters. Siehe Bild:



- Für die Pfadplanung, wäre eine Funktion hilfreich, die zu einer gegebenen Wunschdrehung $\Delta\varepsilon$, eine entsprechende Abfolge von Roboter-Konfigurationen / Posen ausgibt, sodass sich der Roboter entsprechend dreht.

- So könnte zB die Richtung des Roboters so justiert werden, dass er sich auf ein gegebenes Ziel zu bewegt.

- Für den geraden Gang ist eine analytische Funktion bekannt, die die Geschwindigkeit des Roboters einstellt. Geschwindigkeit im Sinne von Schrittweite, bzw. Vorschub pro Zyklus:

$$\boldsymbol{\alpha} = \begin{bmatrix} 45 - \frac{x_1}{2} \\ 45 + \frac{x_1}{2} \\ x_1 \\ 45 - \frac{x_1}{2} \\ 45 + \frac{x_1}{2} \end{bmatrix} \tag{1}$$

Die Schrittweite ist hier als x_1 beschrieben.

2 Approach: Guess structure for a analytic model for walking curves

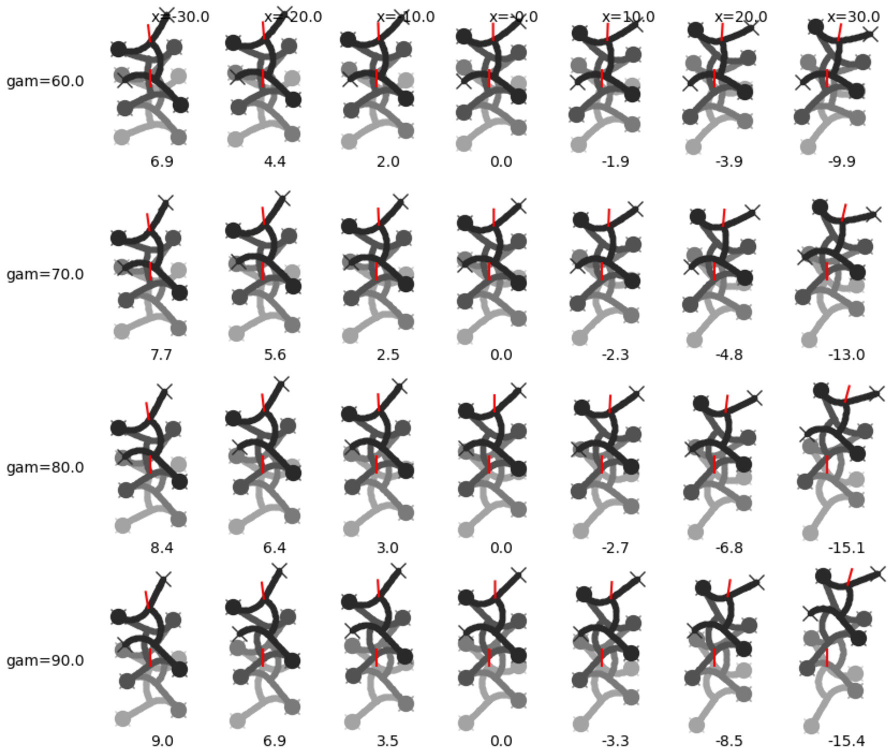
- Src can be found: `analytic_model.py`
- Model:

$$\boldsymbol{\alpha} = \begin{bmatrix} 45 - \frac{x_1}{2} \\ 45 + \frac{x_1}{2} \\ x_1 + x_2 \\ 45 - \frac{x_1}{2} \\ 45 + \frac{x_1}{2} \end{bmatrix} \tag{2}$$

- Method:

  Simulate for different x_1 and x_2 (in der Abbildung unten ist $x_1 =$ `gam` und $x_2 =$ `x`)

- Results für 2 Zyklen:



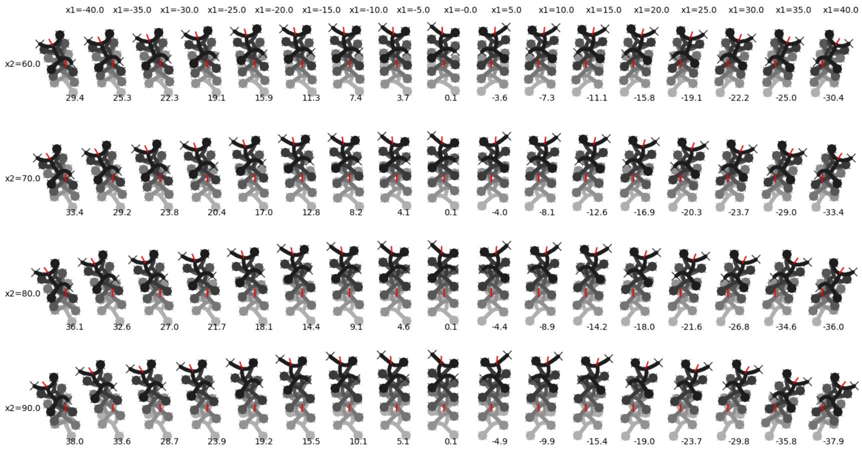
- Observations:
  - Es funktioniert. Der Roboter läuft eine Kurve.
  - Kurve ist unsymmetrisch. Rechts klappt besser als links.
  - Startpose ist besser für Rechtskurve geeignet.
  - Noch nichts über die innere SPannung des Roboters herausgefunden

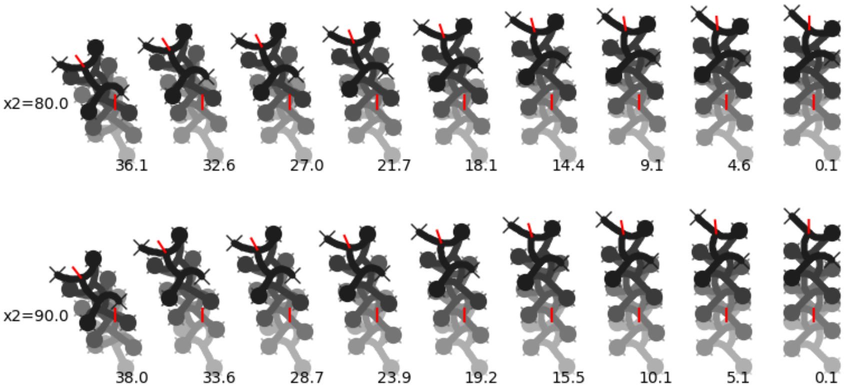
3 Approach: Try another structure

- Src can be found: `analytic_model_2.py`
- Model:

$$\boldsymbol{\alpha} = \begin{bmatrix} 45 - \frac{x_1}{2} + \bar{f}_0|x_1 x_2| + f_0|x1|x_2 \\ 45 + \frac{x_1}{2} + \bar{f}_1|x_1 x_2| + f_1|x1|x_2 \\ x_1 + |x_1|x_2 \\ 45 - \frac{x_1}{2} + \bar{f}_2|x_1 x_2| + f_2|x1|x_2 \\ 45 + \frac{x_1}{2} + \bar{f}_3|x_1 x_2| + f_3|x1|x_2 \end{bmatrix} \tag{3}$$

- Results





4 Approach: Optimize Extra leg bending Angle for given extra torso bending

- Src can be found: `analytic_model_3.py`
- Model:

$$\boldsymbol{\alpha} = \begin{bmatrix} 45 - \frac{x_1}{2} + \bar{f}_0 x_3 + f_0 x_4 \\ 45 + \frac{x_1}{2} + \bar{f}_1 x_3 + f_1 x_4 \\ x_1 + x_2 \\ 45 - \frac{x_1}{2} + \bar{f}_2 x_4 + f_3 x_3 \\ 45 + \frac{x_1}{2} + \bar{f}_3 x_4 + f_4 x_3 \end{bmatrix} \tag{4}$$

- Annahme:

  Die Extra Biegung x_3 für freie Beine und die Extra Biegung x_4 für fixierte Beine sind abhängig von der Extra Biegung x_2 für den Torso.

- Methode:

  Für gegebenes Extra Torso Bending x_2 und gegebenene Torso Biegung $x1$ minimiere die Innere Spannung über den Gang mit n Zyklen aufsummiert:

Gegeben:	x_1	Torsobiegung
	x_2	Extra Torsobiegung
Gesucht:	x_3	Extra Beinbiegung fixiert vorn
	x_4	Extra Beinbiegung fixiert hinten

$$cost(\boldsymbol{x}) = \sum gait(\boldsymbol{x}).stress \tag{5}$$

- Observations:

  Hinter- und Vorderbeine sind nicht symmetrisch, aber kreuzweise symmetrisch: Die Extrabiegung für ein **nicht fixiertes Vorderbein** entspricht der Extrabiegung eines **fixierten Hinterbeins** und anderesherum.