

Algorithm for predicting the pose of a gecko-inspired soft robot for a given reference input

Lars Schiller

June 22, 2018

Due to the fact that the behaviour of soft materials is difficult to predict with conventional methods, an algorithm based on a geometric optimization problem is presented. The algorithm can be used to predict the actual pose of the robot for a given reference pose.

In order to let the robot take a pose, i. e., a position and orientation, the user has nine degrees of freedom at his disposal: the correspondending pressures p_i for the five deformation angles α_i of the limbs $i = 1, \dots, 5$ and the state of the fixation actuators $f_j \in \{0, 1\}$ of the feet $j = 1, \dots, 4$. Accordingly, a reference pose can be described by

$$\mathbf{ref} = [p_1 \ p_2 \ p_3 \ p_4 \ p_5 \ f_1 \ f_2 \ f_3 \ f_4]^\top. \quad (1)$$

Mapping function.

However, this information is not sufficient to describe the robot's actual pose. Experiments have shown that the deformation angle of a limb can vary significantly at the same pressure level due to the softness of the used material. The length of a limb can also differ.

In order to describe the pose of the robot, the deformation angles α_i , the actual lengths of the individual limbs ℓ_i , and the position and orientation of (at least) one foot ($\mathbf{r}_j \in \mathbb{R}^{2 \times 1}$ and φ_j) must be known (see Fig. 1). For example, if the position and orientation of the front left foot is known, the pose can be described by the following expression:

$$\boldsymbol{\alpha} = [\alpha_1 \ \alpha_2 \ \alpha_3 \ \alpha_4 \ \alpha_5]^\top. \quad (2)$$

$$\boldsymbol{\ell} = [\ell_1 \ \ell_2 \ \ell_3 \ \ell_4 \ \ell_5]^\top. \quad (3)$$

$$\boldsymbol{\rho}_{F_1} = [\alpha_1 \ \alpha_2 \ \alpha_3 \ \alpha_4 \ \alpha_5 \ \ell_1 \ \ell_2 \ \ell_3 \ \ell_4 \ \ell_5 \ \mathbf{r}_1^\top \ \varphi_1]^\top. \quad (4)$$

$$\boldsymbol{\rho}_{F_1} = [\mathbf{x} \ \mathbf{r}_1]^\top. \quad (5)$$

Here, the pose seems to be independent of the fixed feet. But the dependency lies in the angles of deformation and the actual lengths of the limbs. These must be determined in such a way that the actual positions of the fixed feet correspond to the theoretical positions from the geometric model. Using the equations (10)–(22), the theoretical positions of all feet can be determined, in case the front left foot is fixed. The equations (23)–(35) describe the positions if the front right foot is fixed. The actual positions of the feet are assumed to be the positions from the previous pose. This can be used to define the constraint for the new pose. All newly fixed feet must have the same position as in the previous step:

$$\text{if } f_i : \ F_{i,j} - F_{i,j-1} \stackrel{!}{=} 0, \quad (6)$$

where i indicates the individual limbs and j the time sequence. So, the new pose is strongly dependent on where the newly fixed feet were in the prior pose and what orientation they had. It has already been mentioned that the lengths and the deformation angles of the limbs are quite variable. The orientation angles of the fixation actuators (c_i) also have a certain margin. This can be formulated as a objective function:

$$obj(\mathbf{x}) = w_{len} \sqrt{\sum_i (\ell_{n,i} - \ell_i)^2} + w_{ang} \sqrt{\sum_i (\varphi_i - \alpha_i)^2} + w_{ori} \sqrt{\sum_i \text{if } f_i : (c_{i,j-1} - c_{i,j})^2} \quad (7)$$

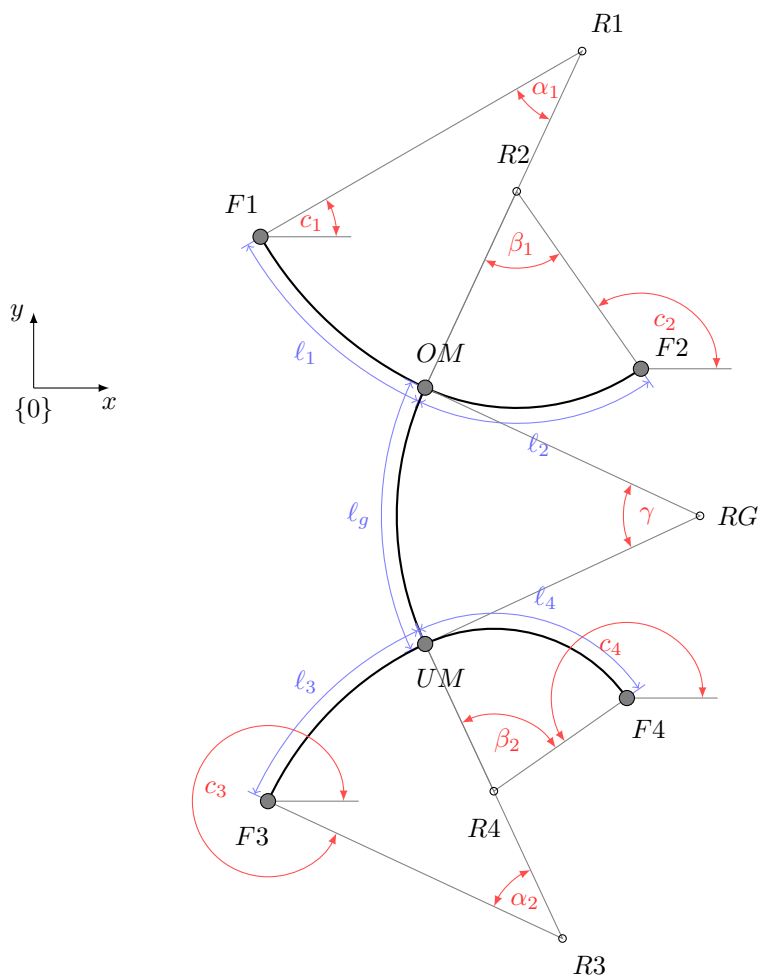


Figure 1: Nomenclature

where the weighting factor w_{len} describes the elasticity of the limbs, w_{ang} the bending stiffness of the limbs and w_{ori} the torsional stiffness of the fixing actuators. Furthermore, $\ell_{n,i}$ describes the nominal length of the i -th limb. The new actual pose can now be determined by solving the optimization problem:

$$\min_{\mathbf{x} \in \mathcal{X}} obj(\mathbf{x}) \quad s.t. \text{ if } f_i : F_{i,j} - F_{i,j-1} = 0. \quad (8)$$

Here $\mathbf{x} = [\alpha_1 \ \beta_1 \ \gamma \ \alpha_2 \ \beta_2 \ c_1 \ c_2 \ \ell_1 \ \ell_2 \ \ell_\gamma \ \ell_3 \ \ell_4]^T$ describes the variables to be optimized, and \mathcal{X} the set of allowed values. Each variable has bounds. For the deformation angles it is:

.....

Radii:

$$r_i = \frac{360 \ \ell_i}{2\pi \ \varphi_i}, \quad \text{for } i \in [1, 2, \gamma, 3, 4] \quad (9)$$

Coordinates for $F1$ fixed:

$$c_2 = c_1 + \alpha_1 + \beta_1 \quad (10)$$

$$c_3 = 180 + \gamma + \alpha_2 + c_1 + \alpha_1 \quad (11)$$

$$c_4 = 180 + \gamma + \alpha_1 - \beta_2 + c_1 \quad (12)$$

$$R1 = \begin{bmatrix} F1_x + \cos(c_1) r_1 \\ F1_y + \sin(c_1) r_1 \end{bmatrix} \quad (13)$$

$$OM = \begin{bmatrix} R1_x - \sin(90 - c_1 - \alpha_1) r_1 \\ R1_y - \cos(90 - c_1 - \alpha_1) r_1 \end{bmatrix} \quad (14)$$

$$RG = \begin{bmatrix} OM_x + \cos(90 - c_1 - \alpha_1) r_\gamma \\ OM_y - \sin(90 - c_1 - \alpha_1) r_\gamma \end{bmatrix} \quad (15)$$

$$UM = \begin{bmatrix} RG_x - \cos(\gamma - 90 + c_1 + \alpha_1) r_\gamma \\ RG_y - \sin(\gamma - 90 + c_1 + \alpha_1) r_\gamma \end{bmatrix} \quad (16)$$

$$R4 = \begin{bmatrix} UM_x + \sin(\gamma - 90 + c_1 + \alpha_1) r_4 \\ UM_y - \cos(\gamma - 90 + c_1 + \alpha_1) r_4 \end{bmatrix} \quad (17)$$

$$F4 = \begin{bmatrix} R4_x + \sin(-c_4 - 90) r_4 \\ R4_y + \cos(-c_4 - 90) r_4 \end{bmatrix} \quad (18)$$

$$R3 = \begin{bmatrix} UM_x + \sin(\gamma - 90 + c_1 + \alpha_1) r_3 \\ UM_y - \cos(\gamma - 90 + c_1 + \alpha_1) r_3 \end{bmatrix} \quad (19)$$

$$F3 = \begin{bmatrix} R3_x - \cos(-c_3) r_3 \\ R3_y + \sin(-c_3) r_3 \end{bmatrix} \quad (20)$$

$$R2 = \begin{bmatrix} OM_x + \cos(c_1 + \alpha_1) r_2 \\ OM_y + \sin(c_1 + \alpha_1) r_2 \end{bmatrix} \quad (21)$$

$$F2 = \begin{bmatrix} R2_x + \sin(c_2 - 90) r_2 \\ R2_y - \cos(c_2 - 90) r_2 \end{bmatrix} \quad (22)$$

Coordinates for $F2$ fixed:

$$c_1 = c_2 - \alpha_1 - \beta_1 \quad (23)$$

$$c_3 = 180 + \gamma - \beta_1 + \alpha_2 + c_2 \quad (24)$$

$$c_4 = 180 + \gamma - \beta_2 + c_2 - \beta_1 \quad (25)$$

$$R2 = \begin{bmatrix} F2_x - \sin(c_2 - 90) r_2 \\ F2_y + \cos(c_2 - 90) r_2 \end{bmatrix} \quad (26)$$

$$OM = \begin{bmatrix} R2_x - \sin(90 - c_2 + \beta_1) r_2 \\ R2_y - \cos(90 - c_2 + \beta_1) r_2 \end{bmatrix} \quad (27)$$

$$RG = \begin{bmatrix} OM_x + \cos(90 - c_2 + \beta_1) r_\gamma \\ OM_y - \sin(90 - c_2 + \beta_1) r_\gamma \end{bmatrix} \quad (28)$$

$$UM = \begin{bmatrix} RG_x - \cos(\gamma - 90 + c_2 - \beta_1) r_\gamma \\ RG_y - \sin(\gamma - 90 + c_2 - \beta_1) r_\gamma \end{bmatrix} \quad (29)$$

$$R4 = \begin{bmatrix} UM_x + \sin(\gamma - 90 + c_2 - \beta_1) r_4 \\ UM_y - \cos(\gamma - 90 + c_2 - \beta_1) r_4 \end{bmatrix} \quad (30)$$

$$F4 = \begin{bmatrix} R4_x + \sin(-c_4 - 90) r_4 \\ R4_y + \cos(-c_4 - 90) r_4 \end{bmatrix} \quad (31)$$

$$R3 = \begin{bmatrix} UM_x + \sin(\gamma - 90 + c_2 - \beta_1) r_3 \\ UM_y - \cos(\gamma - 90 + c_2 - \beta_1) r_3 \end{bmatrix} \quad (32)$$

$$F3 = \begin{bmatrix} R3_x - \cos(-c_3) r_3 \\ R3_y + \sin(-c_3) r_3 \end{bmatrix} \quad (33)$$

$$R1 = \begin{bmatrix} OM_x + \sin(90 - c_2 + \beta_1) r_1 \\ OM_y + \cos(90 - c_2 + \beta_1) r_1 \end{bmatrix} \quad (34)$$

$$F1 = \begin{bmatrix} R1_x - \sin(90 - c_1) r_1 \\ R1_y - \cos(90 - c_1) r_1 \end{bmatrix} \quad (35)$$