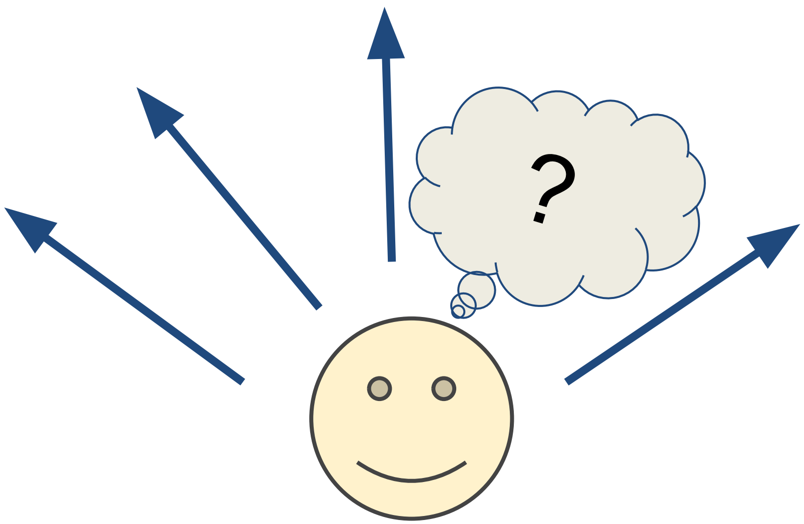
KTH ROYAL INSTITUTE
OF TECHNOLOGY
STOCKHOLM

Learning to make decisions under uncertainty



Judith Bütepage

butepage@kth.se

ROBOTICS, PERCEPTION, AND LEARNING (RPL), CSC, KTH Royal Institute of Technology

Joint work with Ali Ghadirzadeh

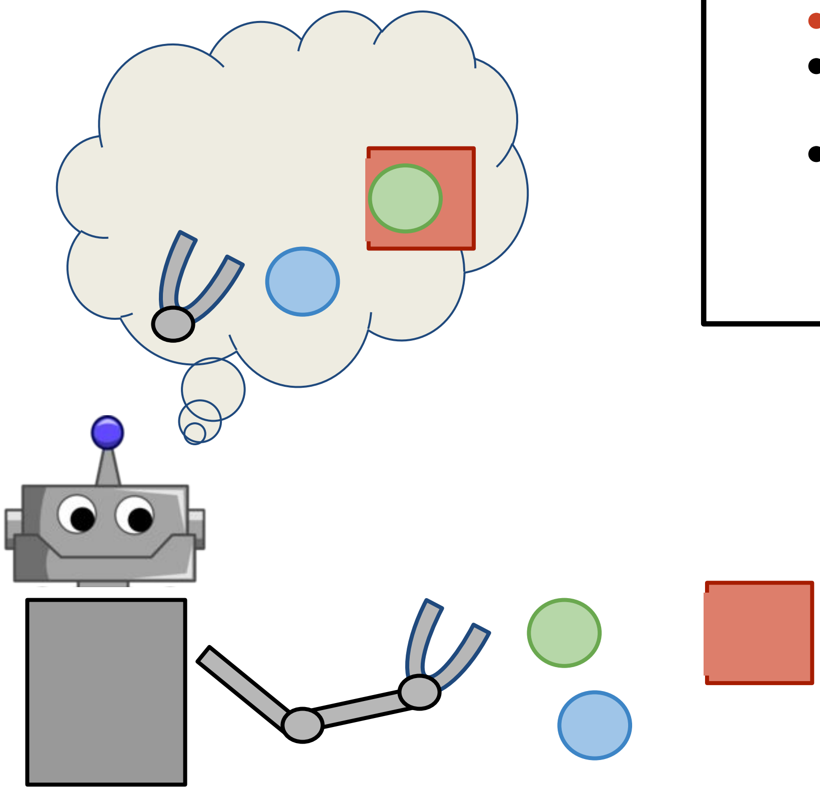
Supported by the EU project – socialized sensorimotor contingencies (socSMCs)

Rational agents

A **rational agent**

- has clear preferences
- models uncertainty via expected values of variables or functions of variables
- chooses actions with the optimal expected outcome for itself from among all feasible actions

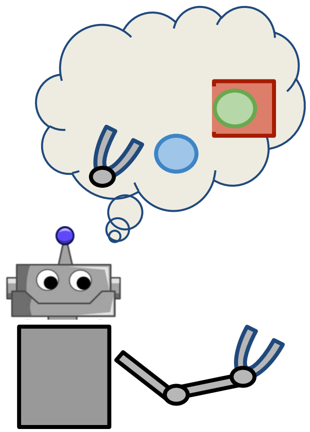
Rational agents



A **rational agent**

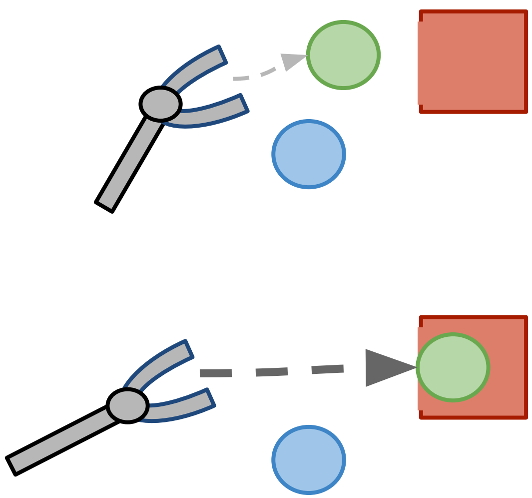
- has clear preferences
- models uncertainty via expected values of variables or functions of variables
- chooses actions with the optimal expected outcome for itself from among all feasible actions

Rational agents



A **rational agent**

- has clear preferences
- models uncertainty via expected values of variables or functions of variables
- chooses actions with the optimal expected outcome for itself from among all feasible actions



$\mathbf{E}$(ball position)

$\mathbf{E}$(ball velocity | push gently) $= \int v \, \mathbf{p}(v \mid pg) \, \partial v$

$\mathbf{E}$(ball velocity | push hard)

$\mathbf{E}$(ball position | old position, velocity)

.

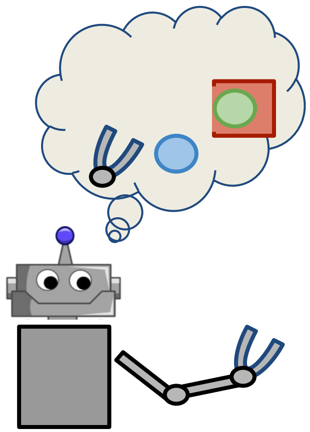
.

.

expectation

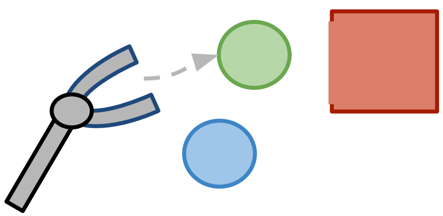
probability density

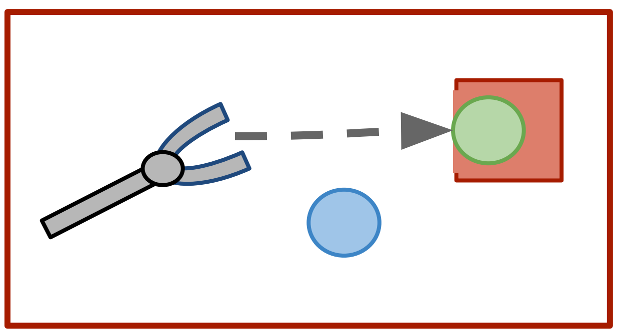
Rational agents



A **rational agent**

- has clear preferences
- models uncertainty via expected values of variables or functions of variables
- chooses actions with the optimal expected outcome for itself from among all feasible actions

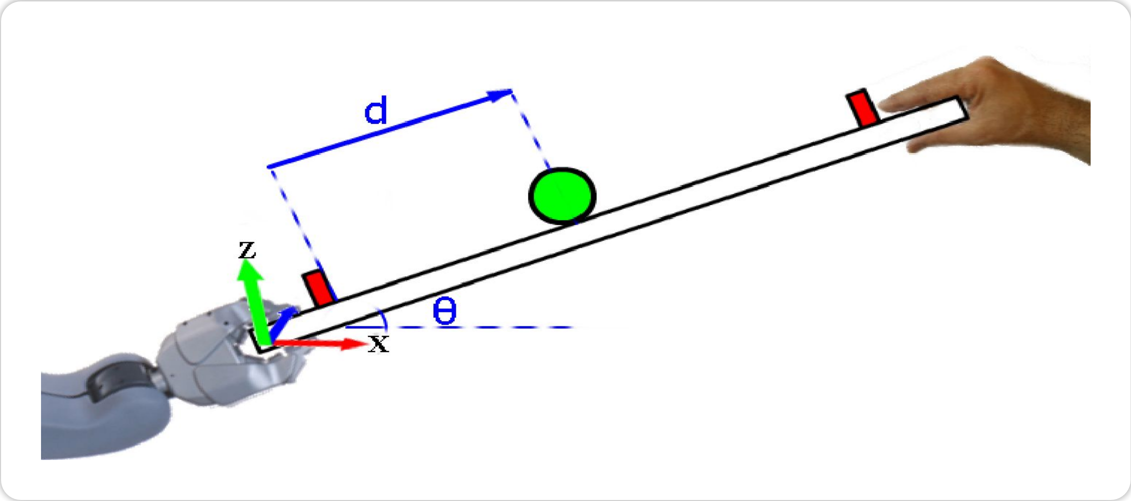




$$\text{action}^* = \underset{\text{action}}{\operatorname{argmax}}\ \mathbf{E}[\ \mathbf{r}(\text{ball position})\ \mathbf{p}(\text{ball position} \mid \text{action})\ ]$$

reward
(determined by
preferences)

Rational agents



A **rational agent**

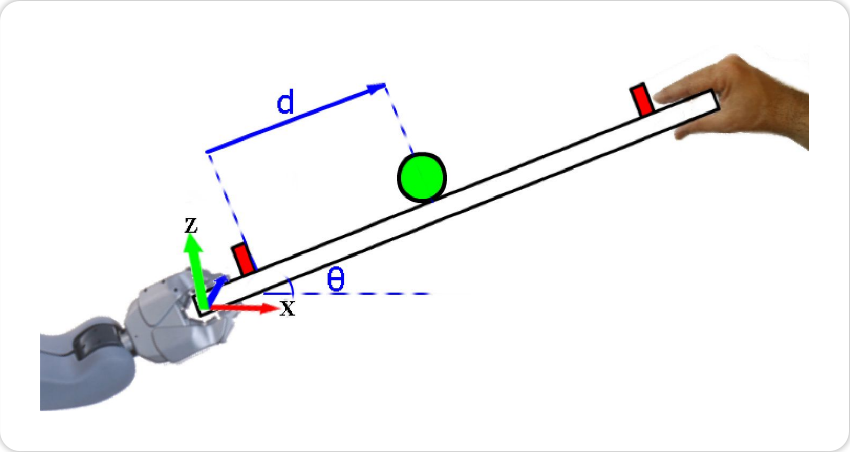
- has clear preferences
- models uncertainty via expected values of variables or functions of variables
- chooses actions with the optimal expected outcome for itself from among all feasible actions

Internal forward models

A forward model predicts future states given current state and action

$$\Delta s_t = \mathscr{F}(s_t, a_t)$$

$$s_t = [x_t, z_t, \Delta d_t, d_t, \dot{d}, \tau_t]$$
$$a_t = [v_x, v_z, \omega]$$
$$\Delta s_t = s_{t+1} - s_t$$



Internal forward models

A forward model predicts future states given current state and action

$$\Delta s_t = \mathscr{F}(s_t, a_t)$$

Problem:

The function $\mathscr{F}$ is often **unknown** and parameterized functions imply **many assumption**!

Solution:

Be Bayesian! Define a **prior over** $\mathscr{F}$ and integrate over all possible functions!

Gaussian Processes

A Gaussian process is a collection of random variables, any finite number of which have a joint Gaussian distribution.

$$f(\mathbf{X}) \sim \mathcal{N}(\mu(\mathbf{X}), K(\mathbf{X}, \mathbf{X}))$$
$$\mathbf{y} = f(\mathbf{X}) + \mathcal{N}(\mathbf{0}, \sigma \mathbf{I})$$

Common choice for the mean function

$$\mu(\mathbf{X}) = \mathbf{0}$$

Common choice for the Kernel function

$$K(\mathbf{x}_m, \mathbf{x}_n) = \sigma_f \exp(\sum_{j=1}^{d} -\frac{\lambda_j (x_m^j - x_n^j)^2}{2}) + \epsilon^2 \delta_{mn}$$

The likelihood of N sample pairs (x_i, y_i) is given by

$$P(\mathbf{y}|\mathbf{X}) = \int_f P(\mathbf{y}|f, \mathbf{X}) P(f|\mathbf{X})$$
$$= \int_f \mathcal{N}(f(\mathbf{X}), \sigma \mathbf{I}) \mathcal{N}(\mu(\mathbf{X}), K(\mathbf{X}, \mathbf{X}))$$

Zero mean assumption

$$= \prod_i^{N} \mathcal{N}(\mathbf{y}_i | \mathbf{0}, K(\mathbf{X}, \mathbf{X}) + \sigma \mathbf{I})$$

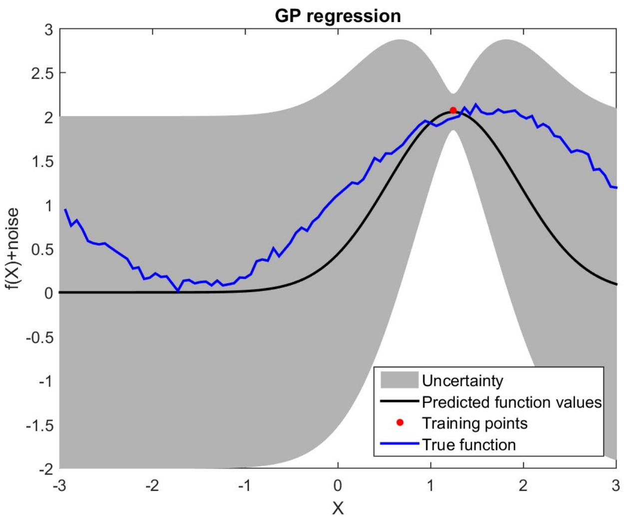
Gaussian Processes

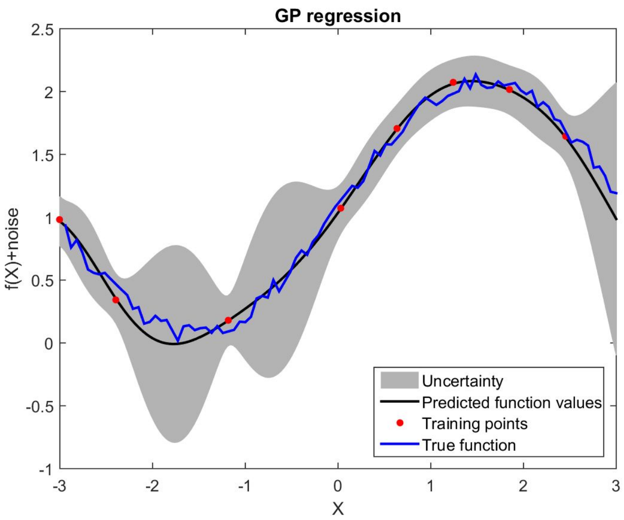
Everything is jointly Gaussian distributed

$$\begin{pmatrix} \mathbf{y} \\ f^* \end{pmatrix} \sim \mathcal{N}\left[\begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix}, \begin{pmatrix} K(\mathbf{X},\mathbf{X}) + \sigma\mathbf{I} & K(\mathbf{X},\mathbf{x}^*) \\ K(\mathbf{x}^*,\mathbf{X}) & K(\mathbf{x}^*,\mathbf{x}^*) \end{pmatrix}\right]$$

Predictive posterior distribution

$$P(f^*|\mathbf{x}^*,\mathbf{X},\mathbf{y}) = \mathcal{N}(m^*,\Sigma^*)$$
$$m^* = K(\mathbf{x}^*,\mathbf{X})K(\mathbf{X},\mathbf{X})^{-1}\mathbf{y}$$
$$\Sigma^* = K(\mathbf{x}^*,\mathbf{x}^*) - K(\mathbf{x}^*,\mathbf{X})K(\mathbf{X},\mathbf{X})^{-1}K(\mathbf{x}^*,\mathbf{X})^T.$$

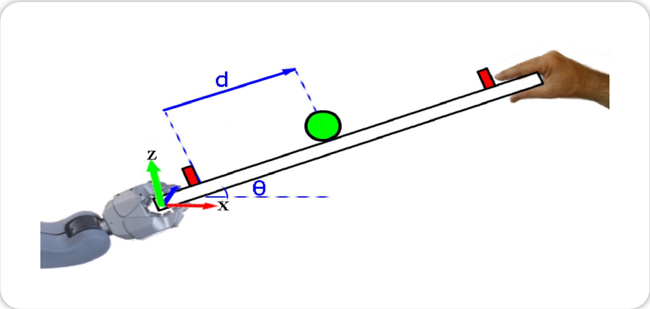




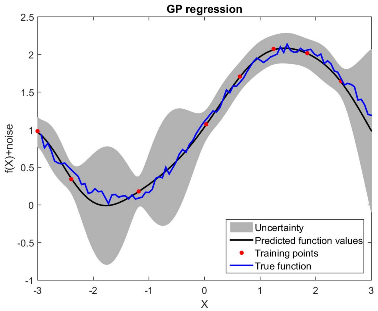
Rational agents

A **rational agent**

- has clear preferences
- models uncertainty via expected values of variables or functions of variables
- chooses actions with the optimal expected outcome for itself from among all feasible actions



$$\Delta s_t = \mathscr{F}(s_t, a_t)$$



Policy Learning

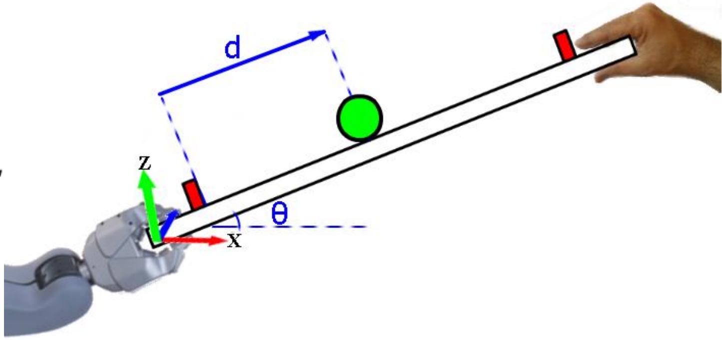
The action-value function (Q-function) represents expected discounted cost for each action-state pair,

$$Q(s_t, a_t) = \mathrm{E}[c(s_{t+1}) + \gamma c(s_{t+2}) + \gamma^2 c(s_{t+3}) + \cdots].$$

The optimal policy is found as:

$$a_{t+1} = \pi(s_t) = argmin[Q(s_t, a_t)]$$

$$s_t = [x_t, z_t, \Delta d_t, d_t, \dot{d}, \tau_t]$$
$$a_t = [v_x, v_z, \omega]$$
$$c(s_t) = [\Delta d_t, \dot{d}_t, \tau] W_c [\Delta d_t, \dot{d}_t, \tau]^T$$



Model-based Gaussian processes Q-learning

Step 1) Predicting a normal distribution over the next state using the forward model

$$\Delta s_t = \mathscr{F}(s_t, a_t)$$

$$s_{t+1} \sim \mathcal{N}(\mu_{t+1}, \Sigma_{t+1})$$

Model-based Gaussian processes Q-learning

Step 1) Predicting a normal distribution over the next state using the forward model

Step 2) Finding the expected cost for the next state

$$\mathbb{E}_{s \sim s_{t+1}}[c(s)] = \int p(s' | \mu_{t+1}, \Sigma_{t+1}) c(s') ds'$$

Model-based Gaussian processes Q-learning

Step 1) Predicting a normal distribution over the next state using the forward model

Step 2) Finding the expected cost for the next state

Step 3) Calculating the expected Q-value over the next state

$$\mathbb{E}_{s \sim s_{t+1}}[Q(s,a)] = \iint p(q|s',a,\theta_q)p(s'|\mu_{t+1},\Sigma_{t+1})q\,ds'dq$$

Model-based Gaussian processes Q-learning

Step 1) Predicting a <u>normal distribution</u> over the <u>next state</u> using the <u>forward model</u>

Step 2) Finding <u>the expected cost</u> for <u>the next state</u>

Step 3) Calculating the <u>expected Q-value</u> over <u>the next state</u>

Step 4) Updating Q-function

$$\boxed{Q(s_t, a_t) \leftarrow \mathbb{E}_{s \sim s_{t+1}}[c(s)] + \gamma \min_{a'} \mathbb{E}_{s \sim s_{t+1}}[Q(s, a')]}$$

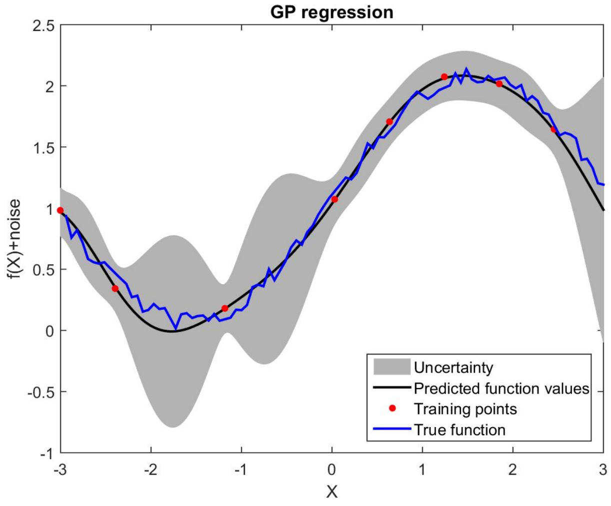
Bayesian optimization for optimal decision making

Bayesian optimization

- **Optimal** action selection
- Staying **close** to the **previous action-state trajectories**

$$Q_{UCB}(x_t) = m_*(x_t) + \delta(k_*(x_t, x_t))^{\frac{1}{2}}$$

$$a_{t+1} = \underset{a^*}{\operatorname{argmin}}\, Q_{UCB}([s_t, a^*])$$

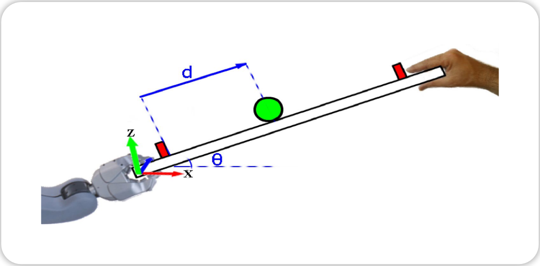


- **Safe exploration**
- **Fast** policy learning
- Action selection under **uncertainty**

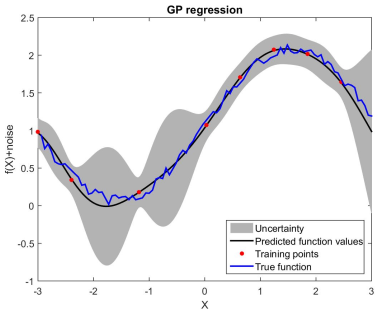
Rational agents

A **rational agent**

- has clear preferences
- models uncertainty via expected values of variables or functions of variables
- chooses actions with the optimal expected outcome for itself from among all feasible actions



$$\Delta s_t = \mathscr{F}(s_t, a_t)$$

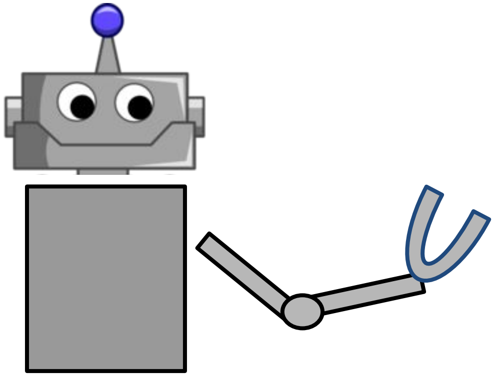


$$Q_{UCB}(x_t) = m_*(x_t) + \delta(k_*(x_t, x_t))^{\frac{1}{2}}$$

$$a_{t+1} = \underset{a^*}{\operatorname{argmin}}\, Q_{UCB}([s_t, a^*])$$

Experiments

Thanks!



Ali Ghadirzadeh

Ghadirzadeh, Ali, et al. "Self-learning and adaptation in a sensorimotor framework." *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE, 2016.

Ghadirzadeh, Ali, et al. "A sensorimotor reinforcement learning framework for physical Human-Robot Interaction." *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*. IEEE, 2016.