Data-efficient reinforcement learning for energy optimization of power-assisted wheelchairs [1][2]

Thierry Marie Guerra
Guoxi Feng,

Lucian Buşoniu

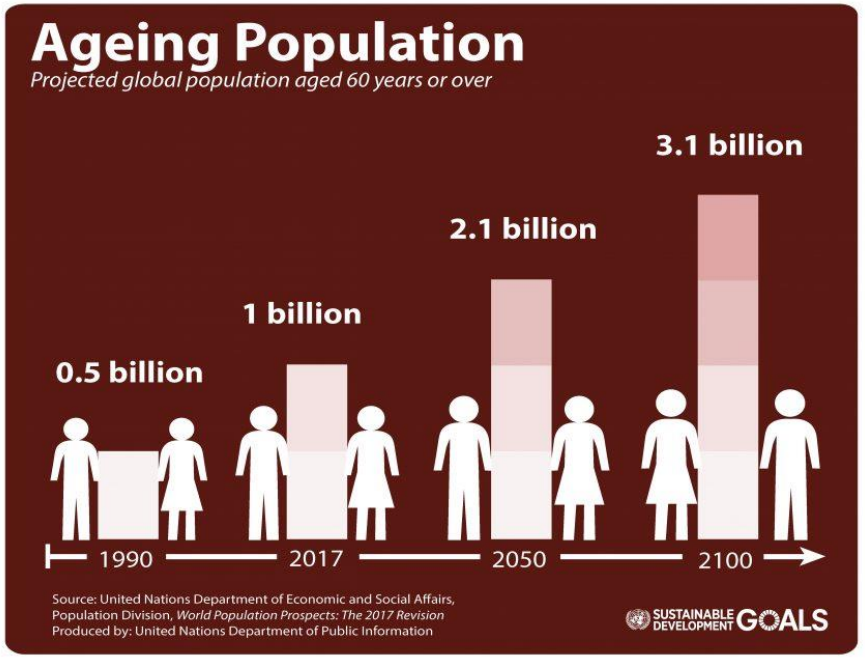
Sami Mohammad CEO Autonomad Mobility

(1) G. Feng, L. Buşoniu, T.M. Guerra, S. Mohammad (2019) – Data-Efficient Reinforcement Learning for Energy Optimization Under Human Fatigue Constraints of Power-Assisted Wheelchairs – IEEE T. on Industrial Electronics, Special Section on: Artificial Intelligence in Ind System, 66 (12), 9734-9744
(2) G. Feng (2019) – Mobility aid for the disabled using unknown input observers and reinforcement learning – PhD Dissertation LAMIH Univ Poytechnique Hauts-de-France

Global disability

- *About 15% of the world's population lives with some form of disability* [2011 world report of the World Health Organization]

- Disabled population is on the rise due to mainly aging



Context: PAWs

- Number of people with reduced mobility is increasing in aging societies.
- Power-assisted wheelchairs (PAWs)

Mohammad et Guerra 2015 Int Patent

PAW: NOMAD & DUO conversion kits that transform a manual wheelchair to an electric wheelchair

Brushless Motors

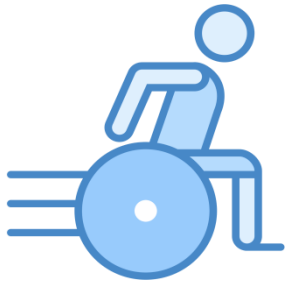
Battery & electronics housing

Control box with joystick

AutoNomad Mobility

Power-Assisted Wheelchair

- Advantages:
  - Human propulsion and electrical motor propulsion.
  - Reduces overuse injuries compared to **manual wheelchairs.**
  - Offers Suitable physical exercise for users compared to **fully electrical wheelchairs.**



Human-wheelchair system

Mechanical wheelchair dynamics

Highly heterogeneous human

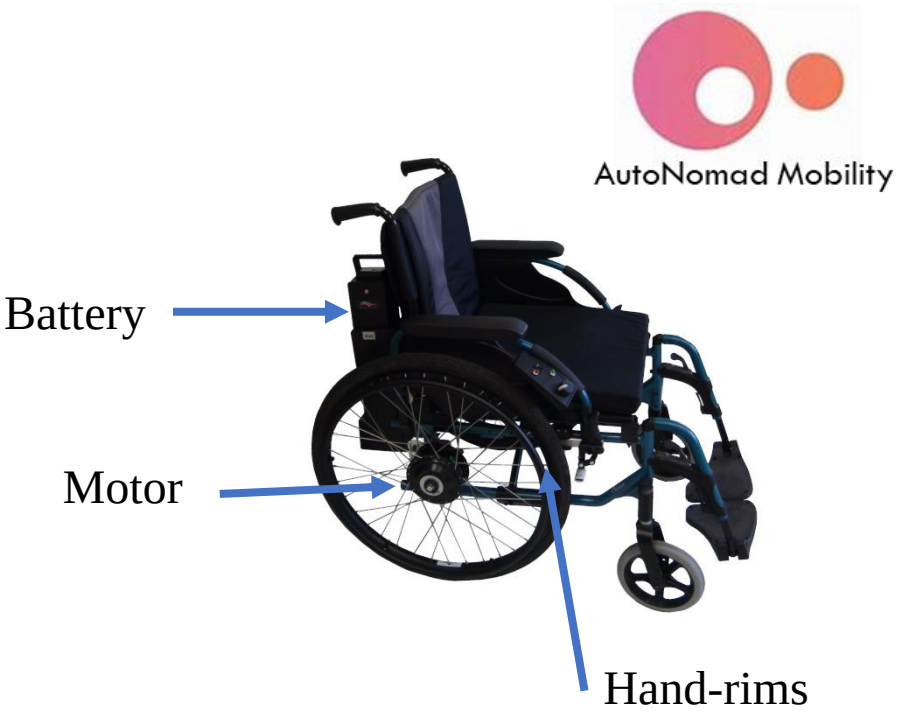
What approach or technique to design assistive control?

What Robust control can do

AutoNomad Mobility

What Robust control can do

- A novel control framework for Power-Assisted Wheelchairs



- With a low manufacturing cost for the larger possible population of disabled persons

Using model-based control and model-free control

- Model-based automatic control enough to deal with:

?

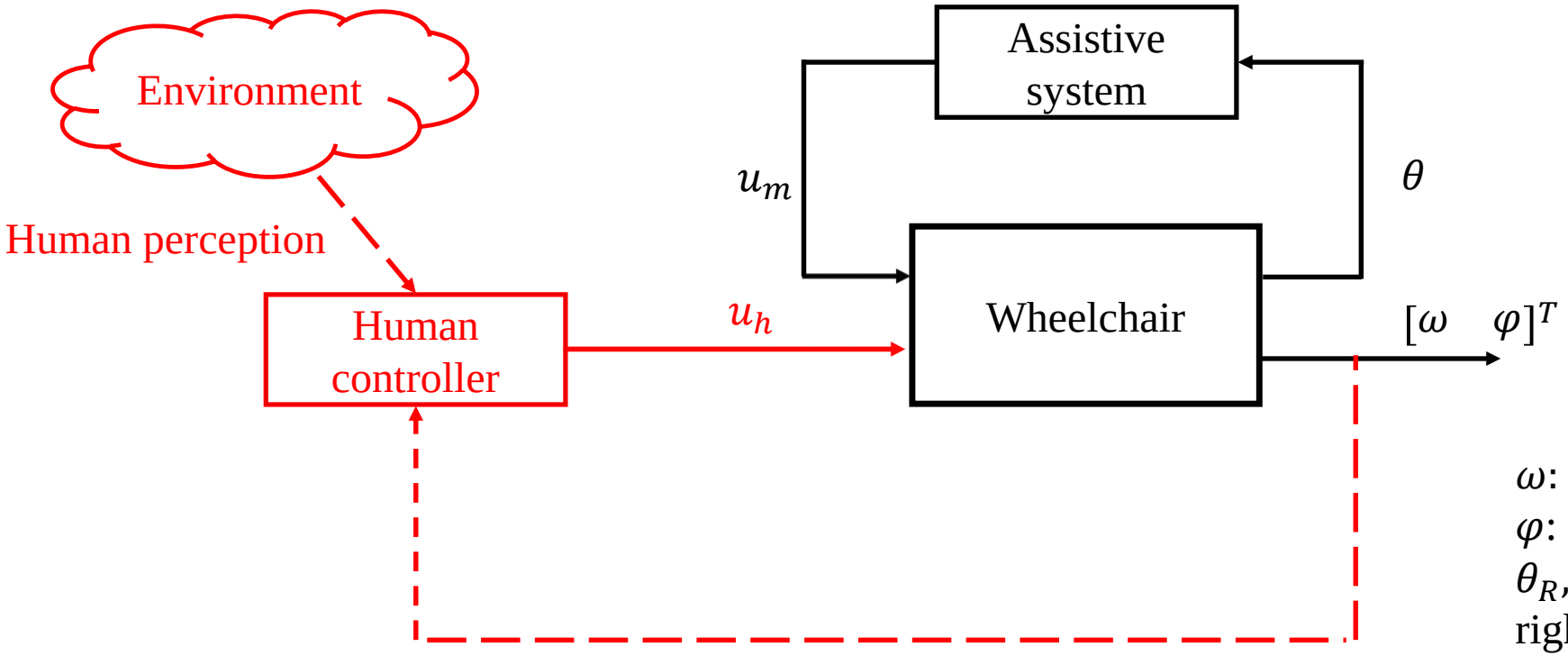
Highly heterogeneous human behaviours

- Modelling precisely the human, every kind of disability and every kind of wheelchair would be infeasible!
- Option: Model-free reinforcement learning.

- $\begin{cases} \text{model-based automatic control} \\ \text{model-free reinforcement learning} \end{cases}$
- Two preliminary studies to see the capabilities and limitations of each approach

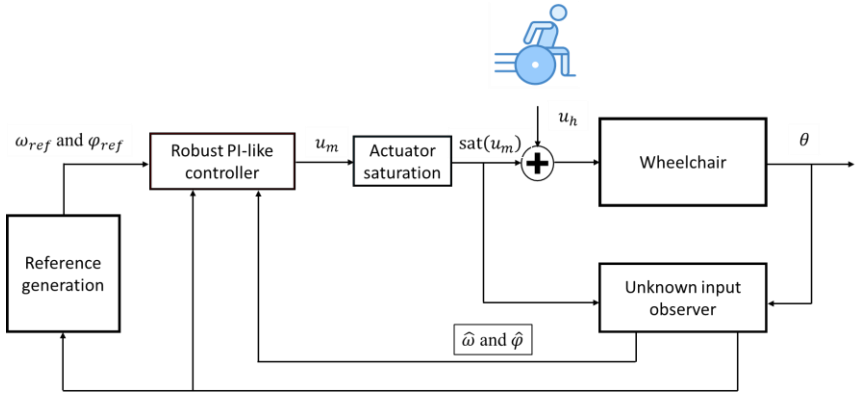
1st study: model-based design for PAWs

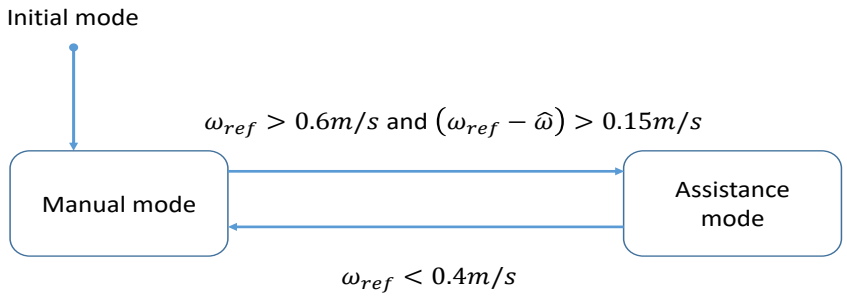
- Objective: Use a simplified model of the wheelchair to develop an assistive control



ω: longitudinal velocity
φ: rotation
θ_R, θ_L: angular velocity of the right/left wheel
u_h: human input
u_m: motor input

Experience with controller+observer





Model-based toward to model-free design

- **Model-based design**
  - ✓ Reconstructs well the human torques
  - ✓ Simplified model and inexpensive encoders
  - ✓ Assistive strategy efficiently helps users to track trajectories
  - ❖ Performance varies between different users
  - ❖ Only robust control strategy but no adaptability to a particular user

- **Model-free design**

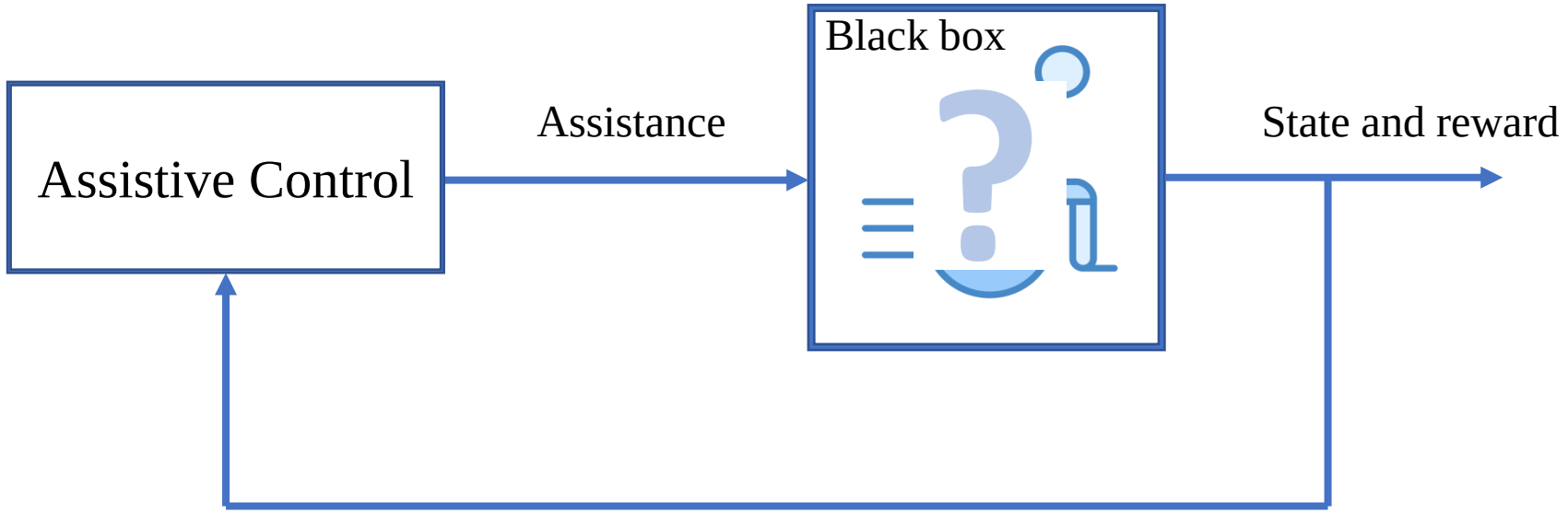
Objective: See if reinforcement learning is able to provide an efficient adaptability for wheelchairs and deal with the high heterogeneous human

Objective with model-free design

- Heterogeneous human behaviors in the control loop

- Compute a (near)optimal strategy without knowing human dynamics:

  Model-free reinforcement learning approach!



Model model free design

Objective: See if reinforcement learning is able to provide an efficient adaptability for wheelchairs and deal with the high heterogeneous human

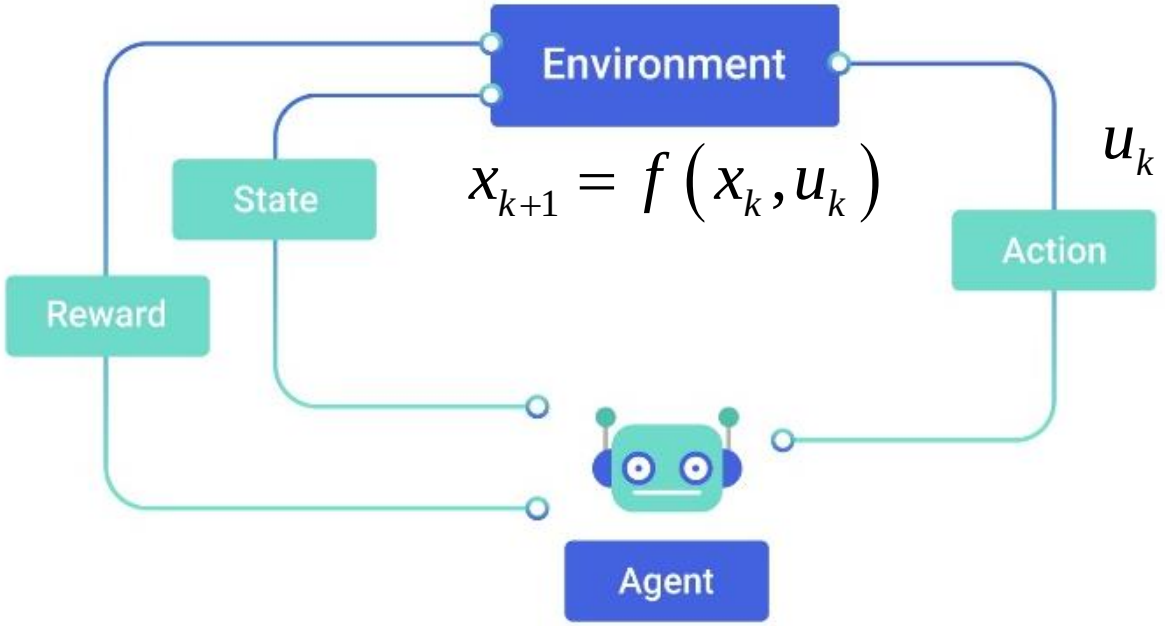
1. Not, just, run and learn
2. “Understand” what happens

Thus: Simple task, from simulation to real-time

The electrical energy consumption over a predefined distance-to-go is optimal, while at the same time bringing users to a desired fatigue level

“Prove things” with models before testing real-time, methodologies = Lucian
Real-time: proof-of-concept

Reinforcement Learning



$$x_{k+1} = f\left(x_k, u_k\right)$$

$$u_k = h\left(x_k\right)$$

Edwards and Fenwick 2016

Reward:

$$r\left(x_k, u_k\right) = E\left(r_{k+1} \middle| x_k = x, u_k = u\right)$$

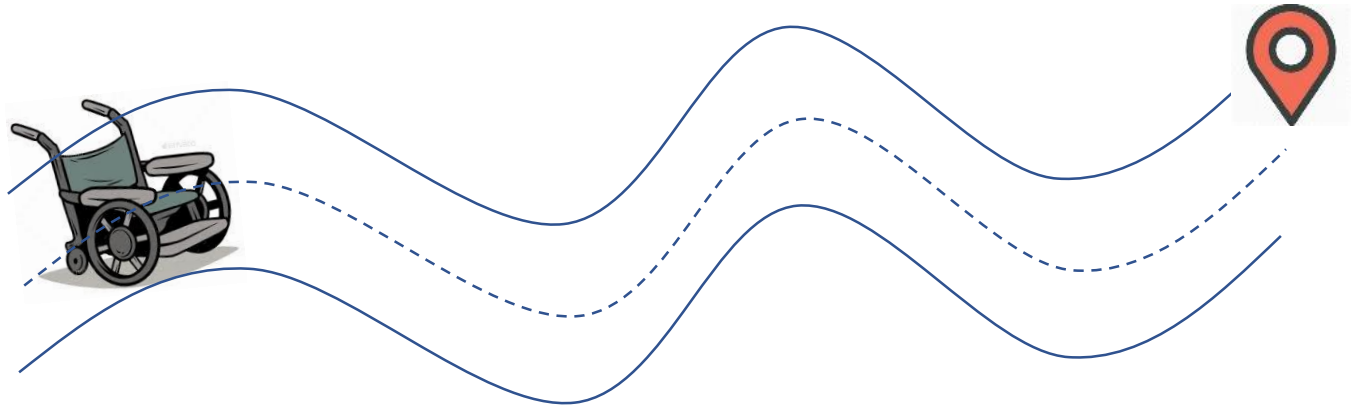
Trajectory: $\tau = \left(x_0 \quad u_0 \quad x_1 \quad u_1 \quad \ldots \quad x_{K-1} \quad u_{K-1} \quad x_K\right)$

General Return: $R\left(\tau\right) = \gamma^K T\left(x_K\right) + \sum_{k=0}^{K-1} \gamma^k r\left(x_k, u_k\right)$

Terminal Reward: $T\left(x_K\right)$

Discount factor: $\gamma \in \left(0,1\right]$

Model-free design: Case study



- Impose distance-to-go or final desired position (knowing initial position):

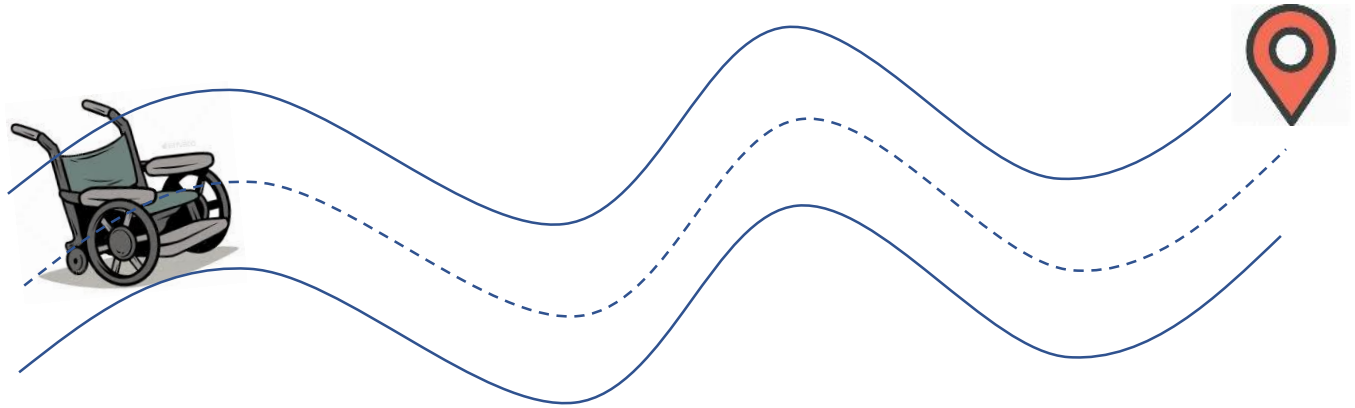
$$\boldsymbol{d(N) = \bar{d}}$$

- Minimize electric energy consumption of motor torque U_m: $\boldsymbol{\frac{1}{2}\sum_{k=0}^{N-1} U_m^2(k)}$

- State of fatigue $S_{of} \in [0,1]$ (avoids trivial solution) desired S_{of} (knowing initial S_{of}):

$$\boldsymbol{S_{of}(N) = \overline{S_{of}}}$$

Finite-Horizon criterion



- Finite-horizon return R to maximize:

$$\max_{U_m} R = \underbrace{-[w_1\ w_2]\begin{bmatrix}\left(d(N)-\bar{d}\right)^2\\ \left(S_{of}(N)-\overline{S_{of}}\right)^2\end{bmatrix}}_{T(x_N)} - \frac{1}{2}\sum_{k=0}^{N-1} U_m^2(k)$$

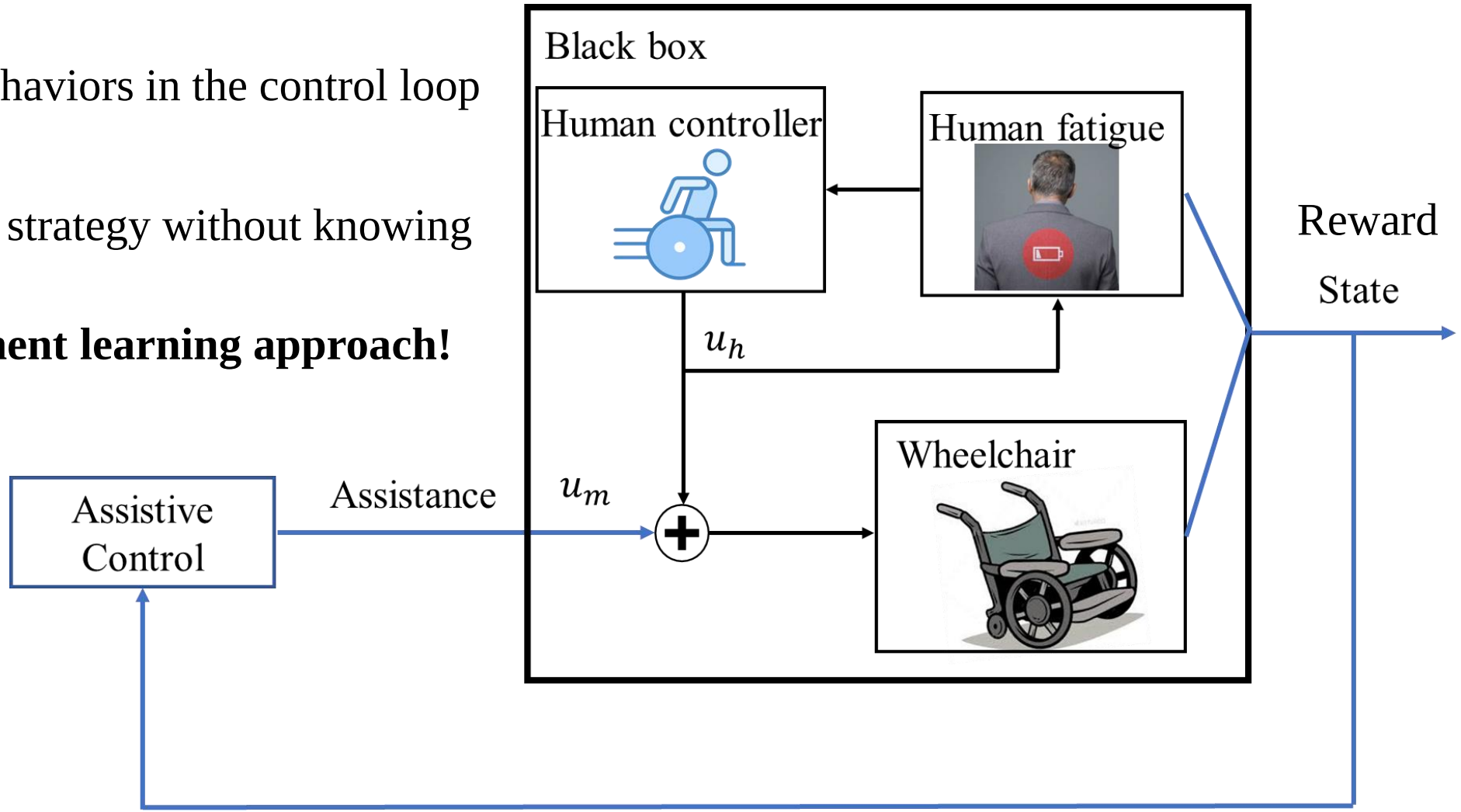
$$r(x_k,u_k)=\frac{1}{2}U_m^2(k)$$

subject to human dynamics and wheelchair dynamics.

Objective with model-free design

- Heterogeneous human behaviors in the control loop
- Compute a (near)optimal strategy without knowing human dynamics:

  Model-free reinforcement learning approach!

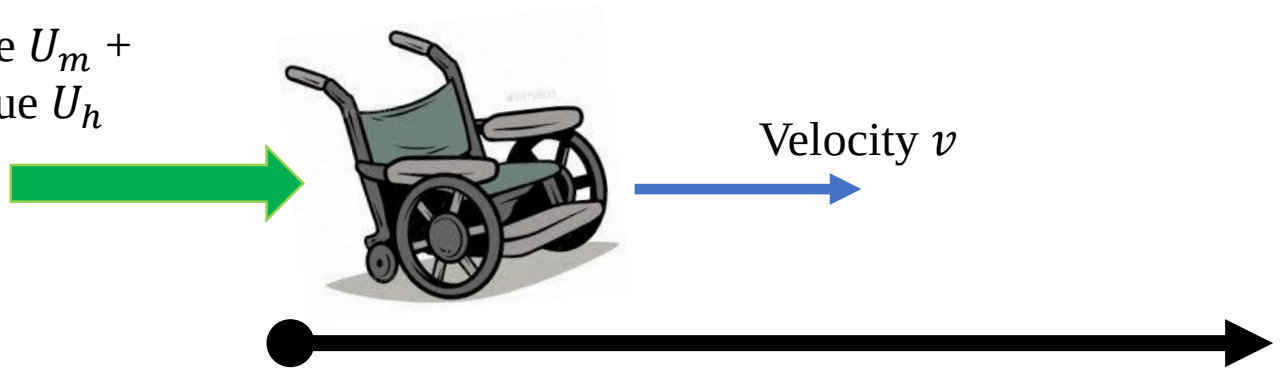


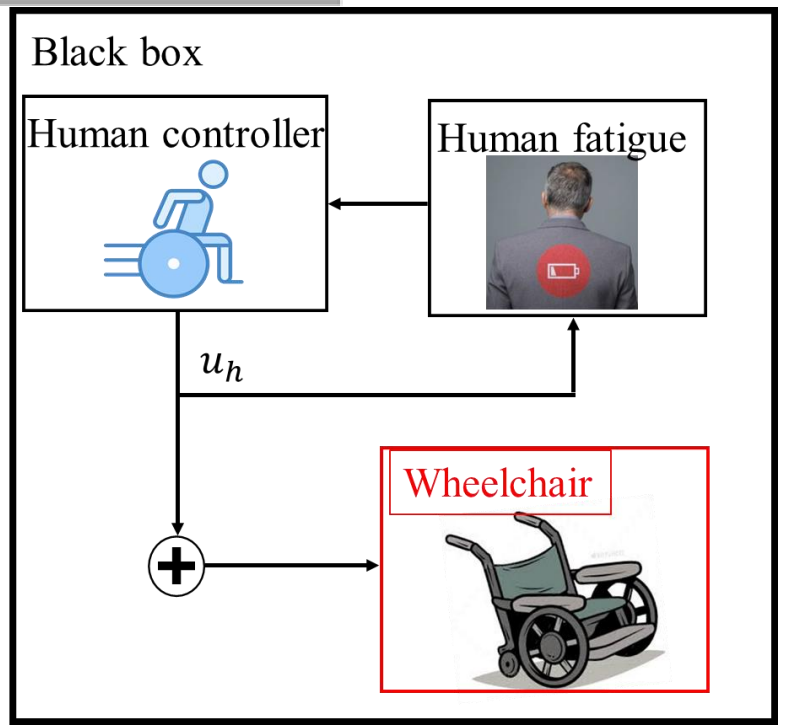
Simulation setup: Coarse modelling

- Wheelchair dynamics:

$$\begin{bmatrix} d(k+1) \\ v(k+1) \end{bmatrix} = \boldsymbol{\mathcal{A}} \begin{bmatrix} d(k) \\ v(k) \end{bmatrix} + \boldsymbol{\mathcal{B}}\big(u_m(k) + u_h(k)\big)$$

Motor torque U_m +
human torque U_h





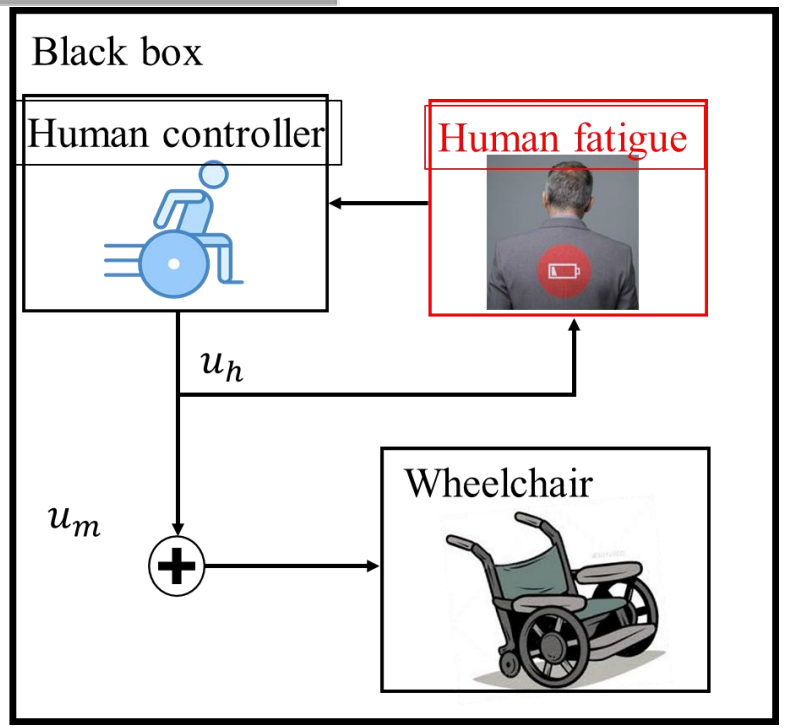
Representative human fatigue

- **Human fatigue dynamics**

Maximum available force F_m: Fayazi, S. 2013

$$\dot{F}_m(t) = -\frac{\mathcal{F}F_h(t)}{M_{vc}}F_m(t) + \mathcal{R}\left(M_{vc} - F_m(t)\right)$$

Fatigue Recovery



F_h: human input; $\mathcal{R}$: Recovery coefficient; $\mathcal{F}$: Fatigue coefficient

State of Fatigue is normalized of F_m :

$$M_{vc} \geq F_m \geq F_{eq}$$

$$\boldsymbol{S_{of}(t)} = \frac{M_{vc} - F_m(t)}{M_{vc} - F_{eq}} \qquad \left(0 \leq S_{of}\ (t) \leq 1\right)$$

Too much exercise $\boldsymbol{S_{of}}$ ↗

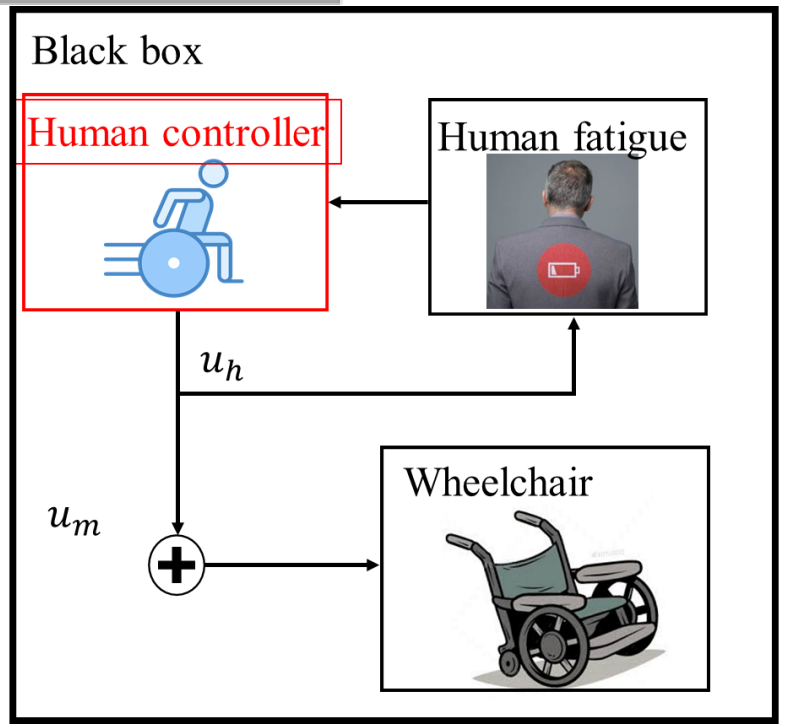
Too little exercise $\boldsymbol{S_{of}}$ ↘

Human Controller

- **Proportional velocity tracking controller:**

$$F_h(k) = K_p\left(V_{max}\mathcal{M} - v(k)\right)$$

- Human motivation $\mathcal{M}\left(S_{of}, U_m\right)$ Ronchi et al. 2016

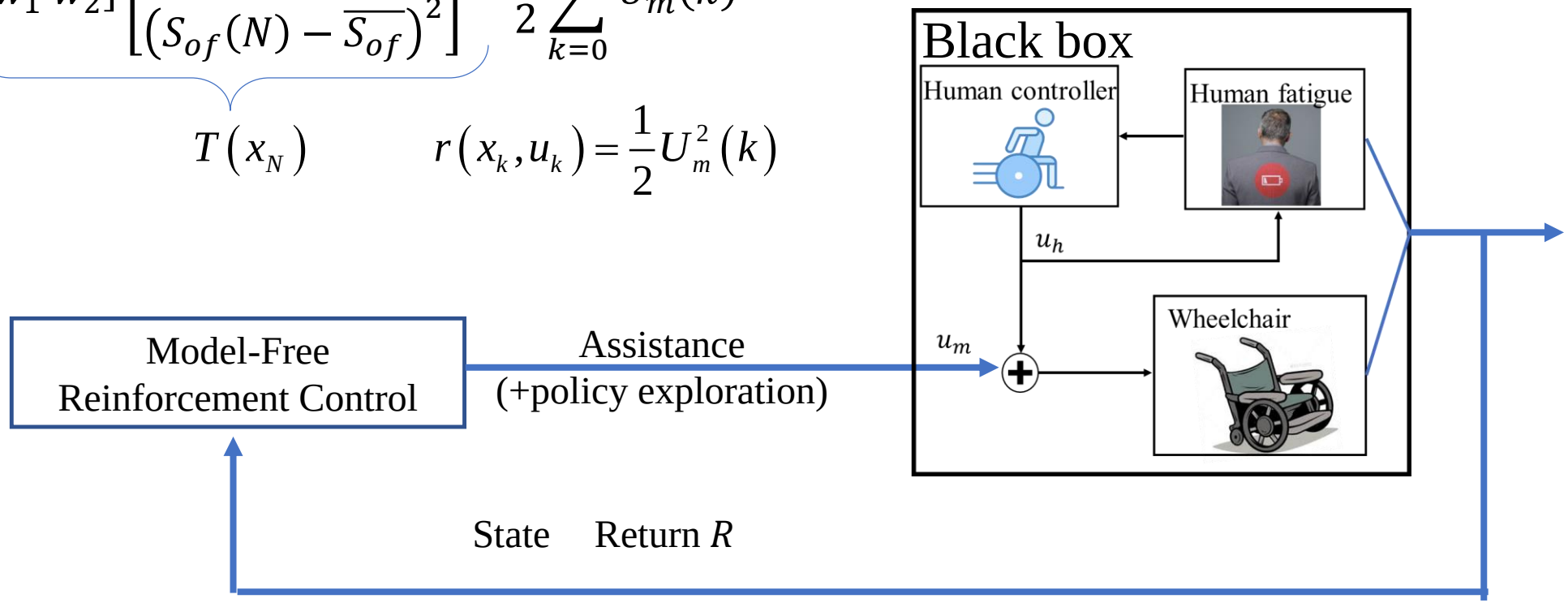


Reinforcement learning

$$x_k = \begin{bmatrix} d_k & v_k & S_{of}(k) \end{bmatrix}^T \quad u_k = U_m(k) \quad \text{motor torque}$$

$$\max_{U_m} R = -\underbrace{[w_1\ w_2]\begin{bmatrix} (d(N)-\bar{d})^2 \\ (S_{of}(N)-\overline{S_{of}})^2 \end{bmatrix}}_{T(x_N)} - \frac{1}{2}\sum_{k=0}^{N-1} U_m^2(k)$$

$$r(x_k, u_k) = \frac{1}{2}U_m^2(k)$$



Baseline solution: Fuzzy Q-iteration

- Q function measures the quality of the state-action pair: Busoniu et al., 2010, Bertsekas et al., 1995

$$Q_k^{\pi}(x_k,u_k)=\gamma^k r(x_k,u_k)+\gamma^{k+1}r(x_{k+1},u_{k+1})+\cdots+\gamma^{K-1}r(x_{K-1},u_{K-1})+\gamma^K T(x_K)$$

for $k=K-1,\ldots,0$ and $\forall x\in X,\ \forall u\in U$

$$\boldsymbol{Q}^* \longrightarrow \boldsymbol{u}^*$$

$$\pi^*(x_k,k)=\arg\max_{u_k} Q_k^*(x_k,u_k)$$

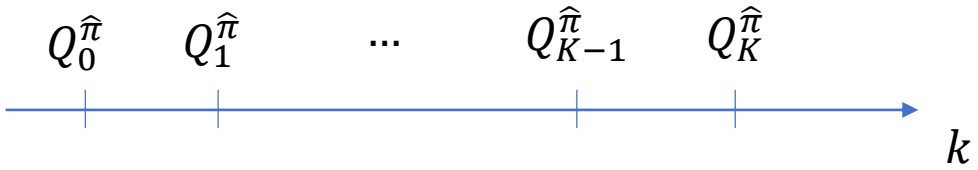
$$Q_{K-1}^*(x_{K-1},u_{K-1})=r(x_{K-1},u_{K-1})+\gamma T\left(\xi(x_{K-1},u_{K-1})\right)$$

$$Q_k^*(x_k,u_k)=r(x_k,u_k)+\gamma\max_{u_{k+1}} Q_{k+1}^*\left(\xi(x_k,u_k),u_{k+1}\right),$$

for $k=K-2,\ldots,0$ and $\forall x\in X,\ \forall u\in U$

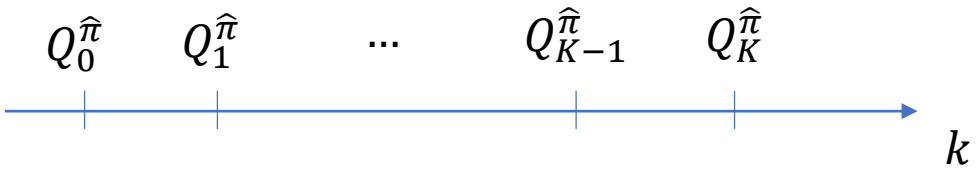
- Use multi-linear interpolations to approximate the optimal time-varying Q function:

$$\hat{Q}_k(x,u)=\sum_{i=1}^{N_x}\phi_i(x)\theta_{i,j,k}$$



Baseline solution: Fuzzy Q-iteration

Bușoniu et al., 2010, Bertsekas et al., 1995



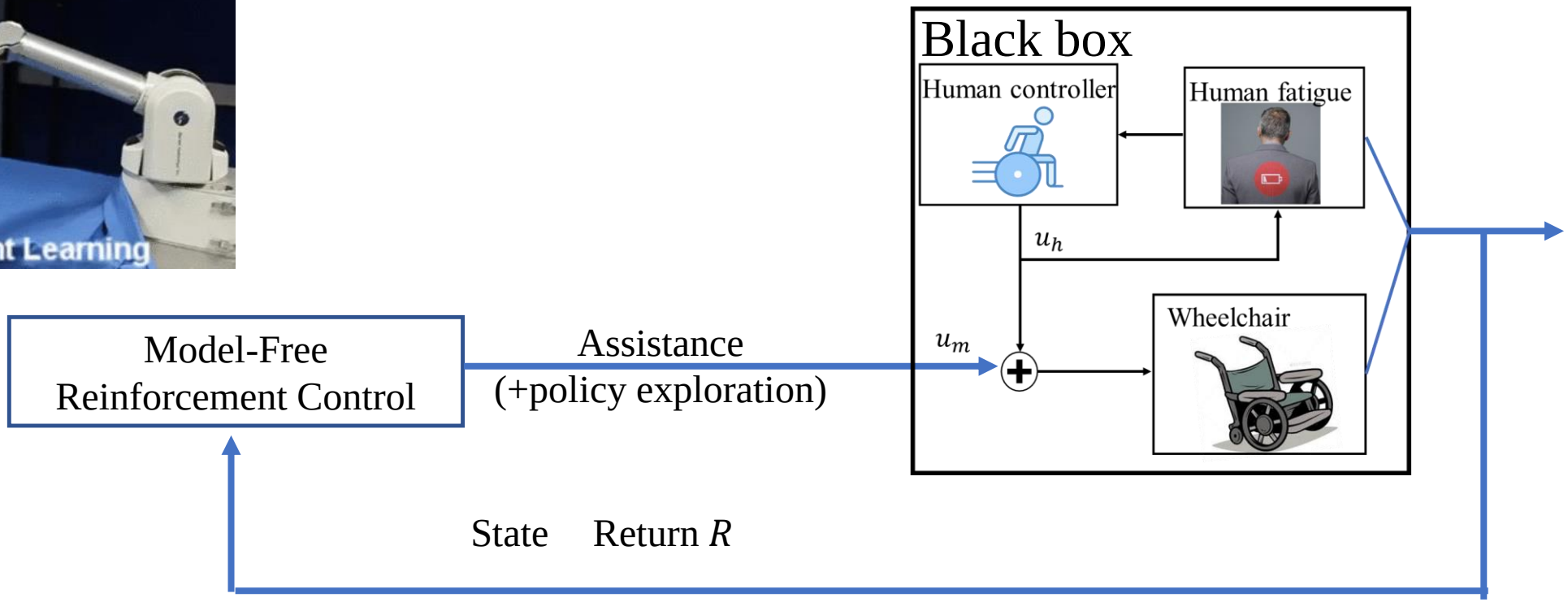
Optimality proof (Finite horizon):

If the reward function r, the terminal function T, and the deterministic state transition function ξ are Lipschitz-continuous,

the approximation error of Q-function is upper bounded by a linear function of the approximation resolution.

Reinforcement learning

- Policy search $\bar{\pi}(x_k) = \lambda_k^T \varphi(x_k), \varphi(x_k) = \left[\varphi_1(x_k) \quad \dots \quad \varphi_M(x_k)\right]$ RBF: $\varphi_i(x_k) = \exp\left(-\beta \left\| x_k - c_i \right\|\right)$ $i \in \{1, \dots M\}$
- Goal: search best control parameters λ^*



GPOMDP: Gradient Partially Observable Markovian Decision Processes

Baxter and Bartlett 2000

Policy search $\bar{\pi}(x_k) = \lambda_k^T \varphi(x_k), \varphi(x_k) = \left[\varphi_1(x_k) \quad \dots \quad \varphi_M(x_k)\right]$ RBF: $\varphi_i(x_k) = \exp\left(-\beta \left\| x_k - c_i \right\|\right)$

$i \in \{1, \dots M\}$

$x_k = \left[d_k \quad v_k \quad S_{of}(k)\right]^T \quad u_k = U_m(k)$ motor torque

GPOMDP: Initialize λ_0

For $l = 0,1,\dots,\Gamma$

Generate N_τ trajectories τ of length K using λ_l

$$\nabla_\lambda \bar{R}_\lambda = \frac{1}{N_\tau} \sum_{\zeta=1}^{N_\tau} \left[\sum_{k=0}^{K-1} \sum_{h=0}^{k} \left(\nabla_\lambda \log \tilde{\pi}_\lambda \left(u_h^\zeta \middle| x_h^\zeta, k \right) \right) r_K^\zeta \right]$$

$\lambda_{l+1} = \lambda_l + \alpha \nabla_\lambda \bar{R}_\lambda, \quad \alpha > 0$ learning rate

End for

For exploration:

$\pi(x_k) \leftarrow \lambda_k^T \varphi(x_k) + z_G$

z_G Gaussian noise

$$\tilde{\pi}_\lambda \left(U_k \middle| x_k \right) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2} \left(U_k - \lambda_k^T \varphi(x_k) \right) \right)$$

GPOMDP: Gradient Partially Observable Markovian Decision Processes

Baxter and Bartlett 2000

Policy search $\bar{\pi}(x_k) = \lambda_k^T \varphi(x_k), \varphi(x_k) = \left[\varphi_1(x_k) \quad \dots \quad \varphi_M(x_k)\right]$ RBF: $\varphi_i(x_k) = \exp\left(-\beta \|x_k - c_i\|\right)$

$i \in \{1, \dots M\}$

$x_k = \left[d_k \quad v_k \quad S_{of}(k)\right]^T \quad u_k = U_m(k)$ motor torque

GPOMDP: Initialize λ_0

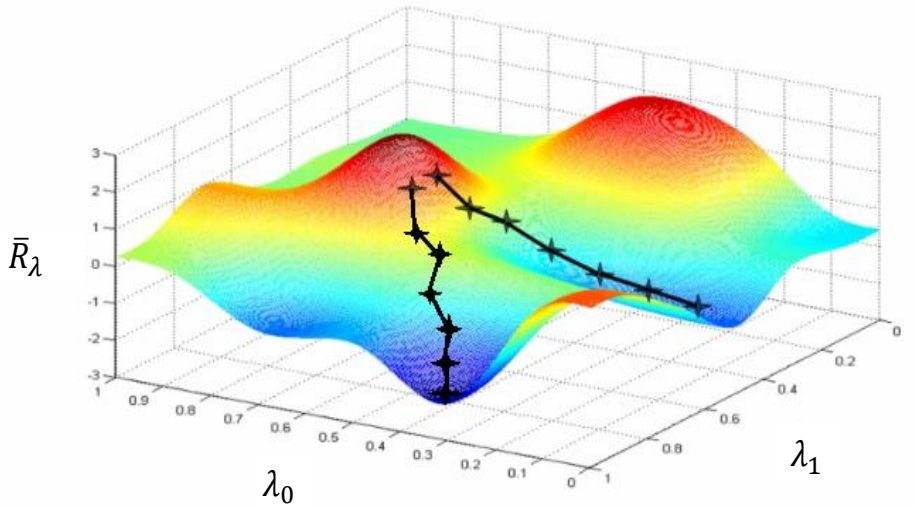
For $l = 0,1,\dots,\Gamma$

Generate N_τ trajectories τ of length K using λ_l

$$\nabla_\lambda \bar{R}_\lambda = \frac{1}{N_\tau} \sum_{\zeta=1}^{N_\tau} \left[\sum_{k=0}^{K-1} \sum_{h=0}^{k} \left(\nabla_\lambda \log \tilde{\pi}_\lambda \left(u_h^\zeta \middle| x_h^\zeta, k \right) \right) r_K^\zeta \right]$$

$\lambda_{l+1} = \lambda_l + \alpha \nabla_\lambda \bar{R}_\lambda, \quad \alpha > 0$ learning rate

End for



GPOMDP: Gradient Partially Observable Markovian Decision Processes

Baxter and Bartlett 2000

Policy search $\bar{\pi}(x_k) = \lambda_k^T \varphi(x_k), \varphi(x_k) = \left[\varphi_1(x_k) \quad \dots \quad \varphi_M(x_k)\right]$

RBF: $\varphi_i(x_k) = \exp\left(-\beta \left\| x_k - c_i \right\|\right)$

$i \in \{1, \dots M\}$

$x_k = \left[d_k \quad v_k \quad S_{of}(k)\right]^T \quad u_k = U_m(k)$ motor torque

GPOMDP: Initialize λ_0

For $l = 0,1,\dots,\Gamma$

Generate N_τ trajectories τ of length K using λ_l

$$\nabla_\lambda \bar{R}_\lambda = \frac{1}{N_\tau} \sum_{\zeta=1}^{N_\tau} \left[\sum_{k=0}^{K-1} \sum_{h=0}^{k} \left(\nabla_\lambda \log \tilde{\pi}_\lambda \left(u_h^\zeta \middle| x_h^\zeta, k \right) \right) r_K^\zeta \right]$$

$\lambda_{l+1} = \lambda_l + \alpha \nabla_\lambda \bar{R}_\lambda, \quad \alpha > 0$ learning rate

End for

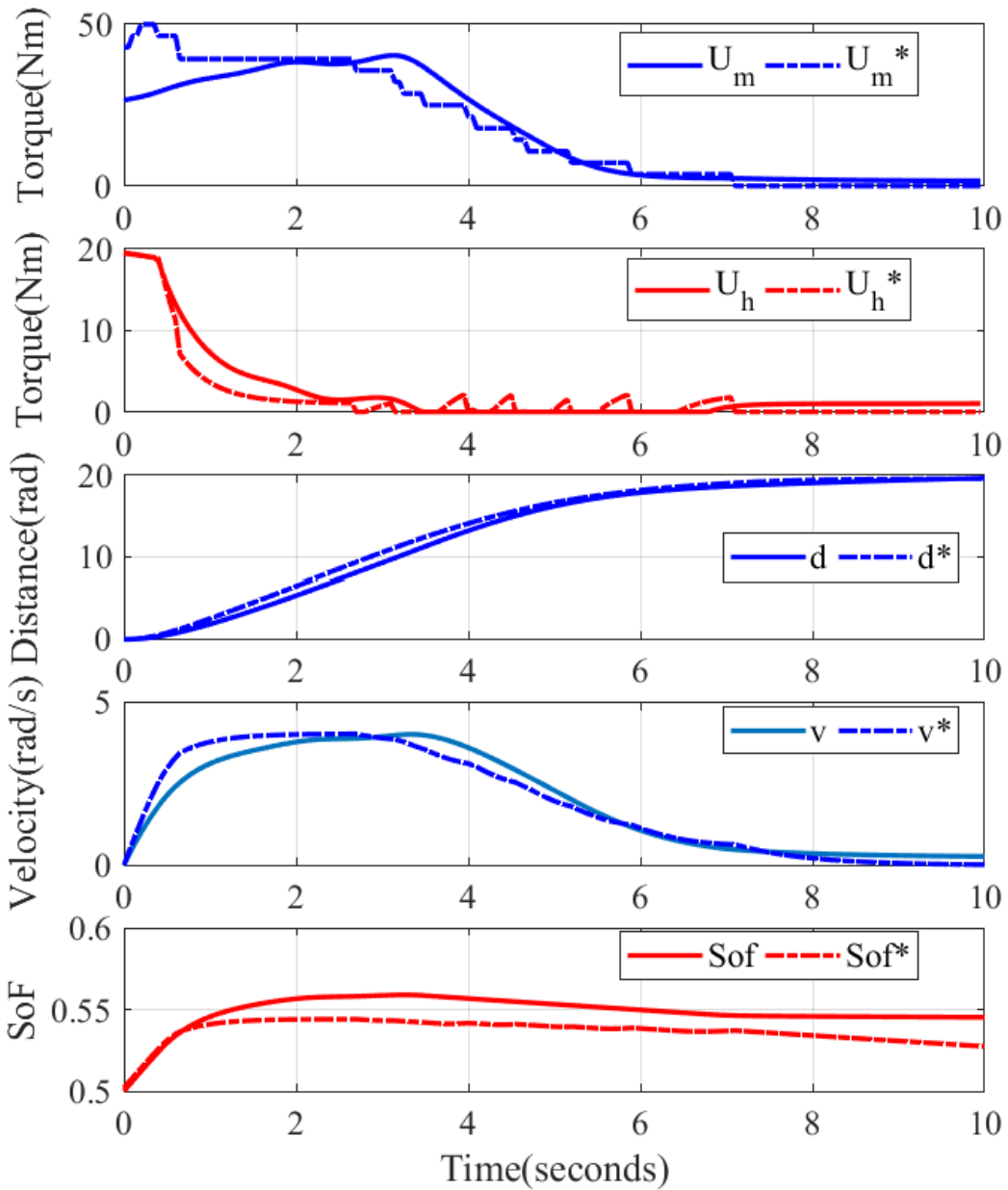
Need to know:

$\beta, c_i, M, i \in \{1, \dots M\}$

σ, w_1, w_2

α, N_τ

GPOMDP: Gradient Partially Observable Markovian Decision Processes



$\left[U_m^*, U_h^* \ldots\right]$ Fuzzy $Q-$iteration

$10s$	Trajectory
$s = 0.05s$	sampling time
$\beta = 0.5,\ 5\times5\times8\ c_i\ (M = 200)$	RBF
σ	5
$[w_1,\ w_2]$	$\begin{bmatrix} 4000 & 10^7 \end{bmatrix}$
$\alpha = 10^{-5}$	learning rate
8000	n° trajectories

PoWER (Policy learning by Weighting Exploration with the Returns)

- A lower bound on the expected rewards: Kober and Peters 2009

Maximize: $L_\lambda(\lambda') = \int p_\lambda(\tau) R(\tau) \log\left(\frac{p_{\lambda'}(\tau)}{p_\lambda(\tau) R(\tau)}\right) d\tau$ $\Longleftrightarrow$ $L_\lambda(\lambda') = -D\big(p_\lambda(\tau) R(\tau) || p_{\lambda'}(\tau)\big)$

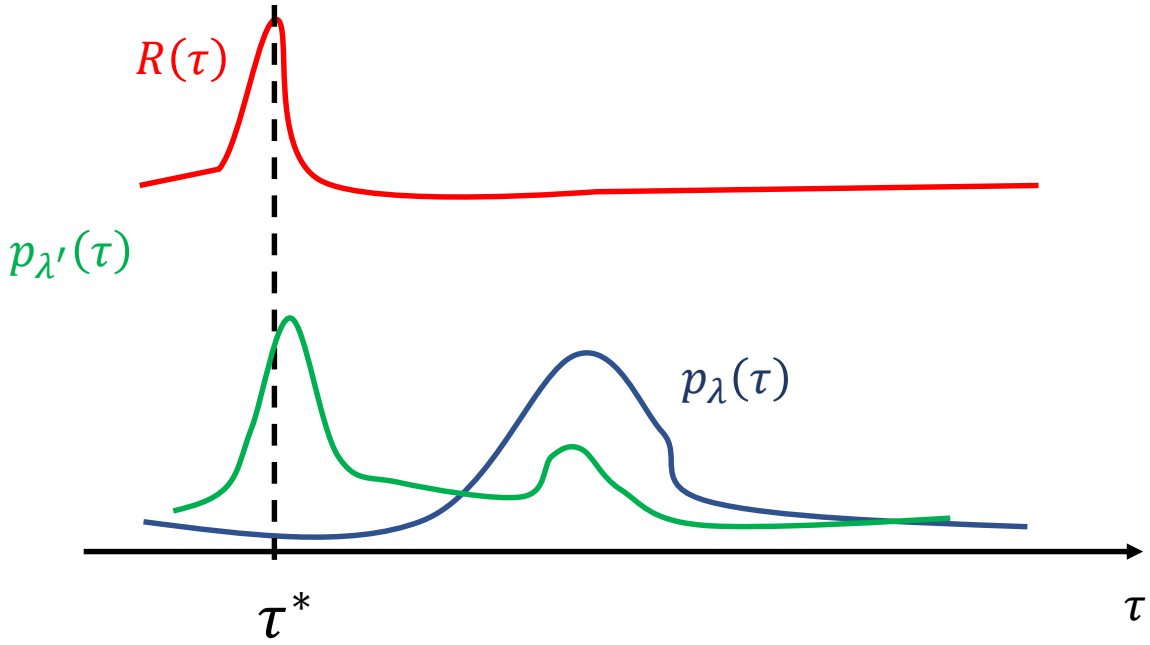
Operator D Kullback–Leibler divergence: Measure the distance of two distributions

Maximize: $-D\big(p_\lambda(\tau) R(\tau) || p_{\lambda'}(\tau)\big)$

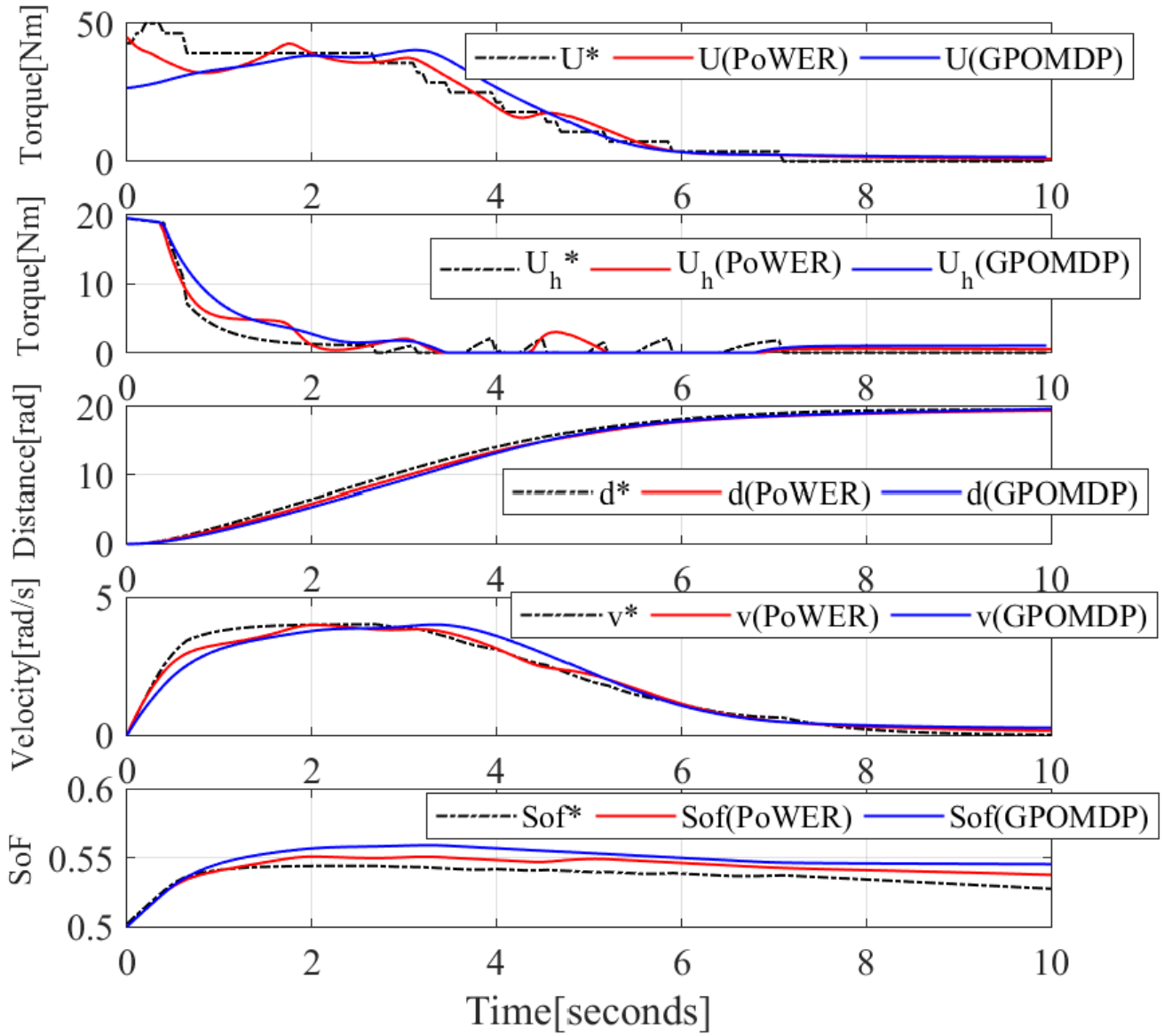
$\Updownarrow$

Minimize the distance of $p_\lambda(\tau) R(\tau)$ and $p_{\lambda'}(\tau)$

$$\lambda_{l+1} = \lambda_l + \frac{\sum_{s=1}^{N_s} (\lambda_s - \lambda_l) R(\tau_s)}{\sum_{s=1}^{N_s} R(\tau_s)}$$



PoWER (Policy learning by Weighting Exploration with the Returns**)**

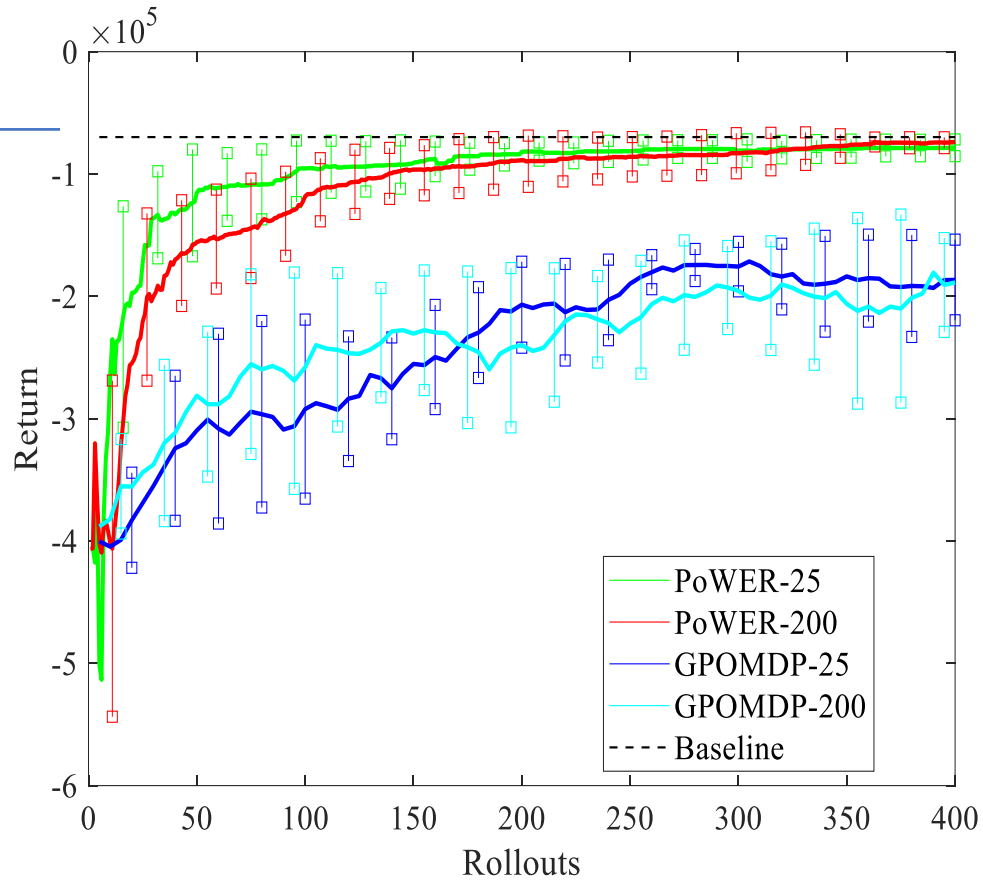


$$\bar{\pi}\left(x_k\right) = \lambda_k^T \psi\left(x_k\right)$$

$10s$	Trajectory
$s = 0.05s$	sampling time
$\beta = 0.5,\ 5\times5\times1\ c_i\ \left(M = 25\right)$	RBF
σ	1
$\left[w_1,\ w_2\right]$	$\left[4000 \quad 10^7\right]$
$N_s = 10$	Importance Sampling
400	n° trajectories

GPOMDP vs PoWER

Fuzzy Q-iteration



- 2 configurations of policy approximation for each approach.
  - 25 basis functions
  - 200 basis functions

- Mean value along with a 95% confidence interval calculated for 10 independent simulations is given (each simulation with 400 trials)

GPOMDP-25 vs PoWER-25

- 400 trials and 8000 trials are performed to learn the parameter vectors λ^P (PoWER-25) and λ^G(GPOMDP-25), respectively.

- Since GPOMDP has a lower data efficiency, more trials are needed to provide a good behaviour.

GPOMDP-25 provides 12.7% less return than Fuzzy Q-iteration!

PoWER-25 has a similar return as Fuzzy Q-iteration and learns much faster.

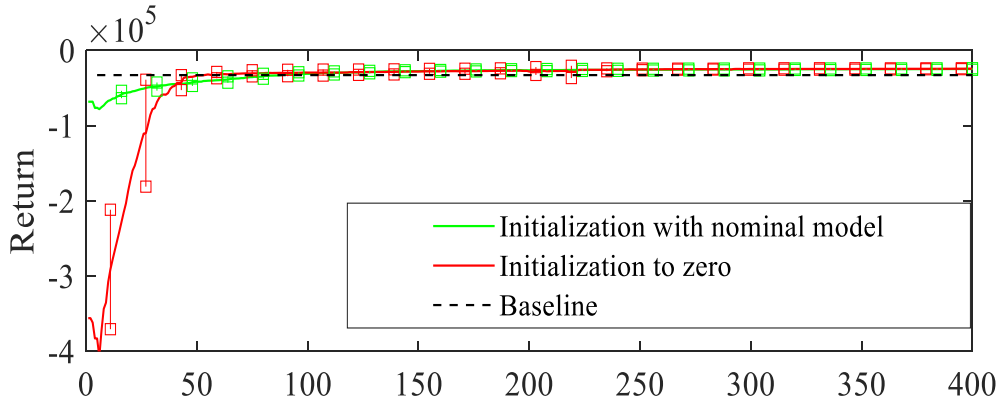
Test the adaptability of PoWER-25 to different fatigue dynamics

Adaptability – PoWER-25

- To represent various human fatigue dynamics:

$$\mathcal{F}' = \frac{1}{\boldsymbol{\eta}}\mathcal{F}; \quad \mathcal{R}' = \boldsymbol{\eta}\mathcal{R}; \quad M'_{vc} = \boldsymbol{\eta} M_{vc} \qquad \boldsymbol{\eta} > 1 \text{ exhausted slower, recover fast, Max force bigger}$$

Stronger $\boldsymbol{\eta} = 2$

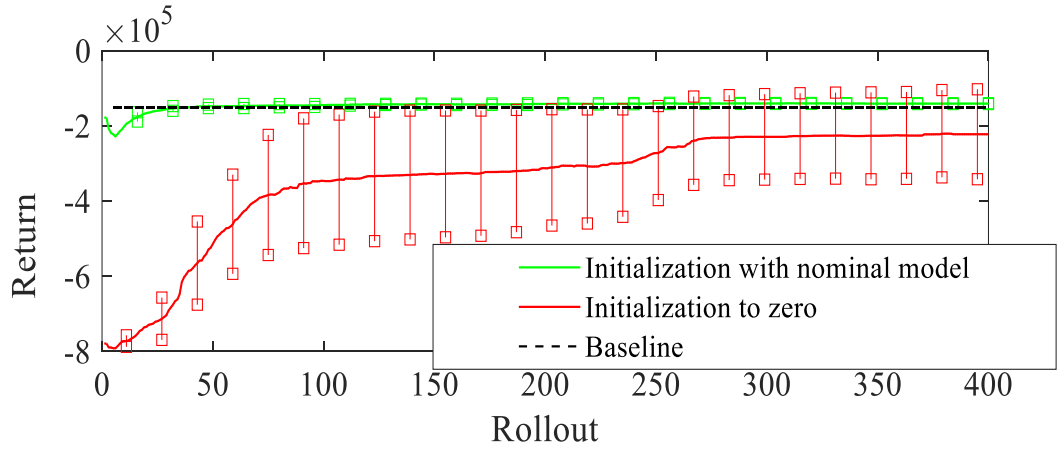


- Different initialization of the policy:

  - Initialized with the parameters learned from the nominal model

  - Initialized to zero by default

Weaker $\boldsymbol{\eta} = \frac{1}{2}$



Adaptability

$$\mathcal{F}' = \frac{1}{\boldsymbol{\eta}}\mathcal{F}; \qquad \mathcal{R}' = \boldsymbol{\eta}\mathcal{R}; \qquad M'_{vc} = \boldsymbol{\eta} M_{vc}$$

- To test different human fatigue dynamics

Zero: Initialization to zero
Nominal: Initialization with the nominal model.

/ represents situations where the learning algorithm fails to converge to the 90% of the baseline return within 400 trials.

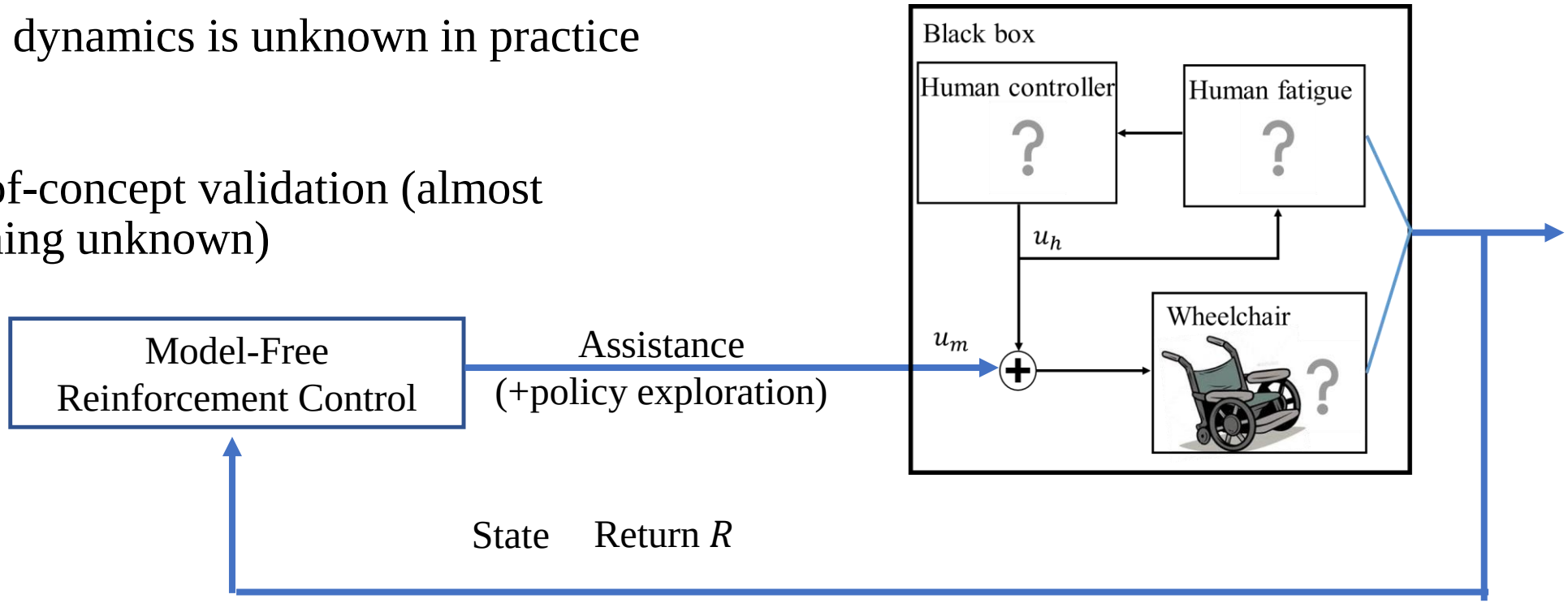
	Number of trials to 90% of baseline return	
η	Nominal	Zero
8	39	37
4	33	56
3	47	34
2	48	65
1/2	10	/
1/3	198	/
1/4	37	/
1/8	30	/

Stronger

Weaker

Simulation towards real world

- Simulations show the efficient adaptability of PoWER regards to different fatigue dynamics
- Human dynamics is unknown in practice
- Proof-of-concept validation (almost everything unknown)



Model-free design: proof-of-concept validation

- User returns his feeling via a joystick $I = \begin{cases} 1 & \text{insufficiently tired (is willing to exercise more)} \\ 0 & \text{comfortable} \\ -1 & \text{too tired} \end{cases}$

- Scenario: follow a given reference velocity v_{ref} (same straight flat road for each trial):

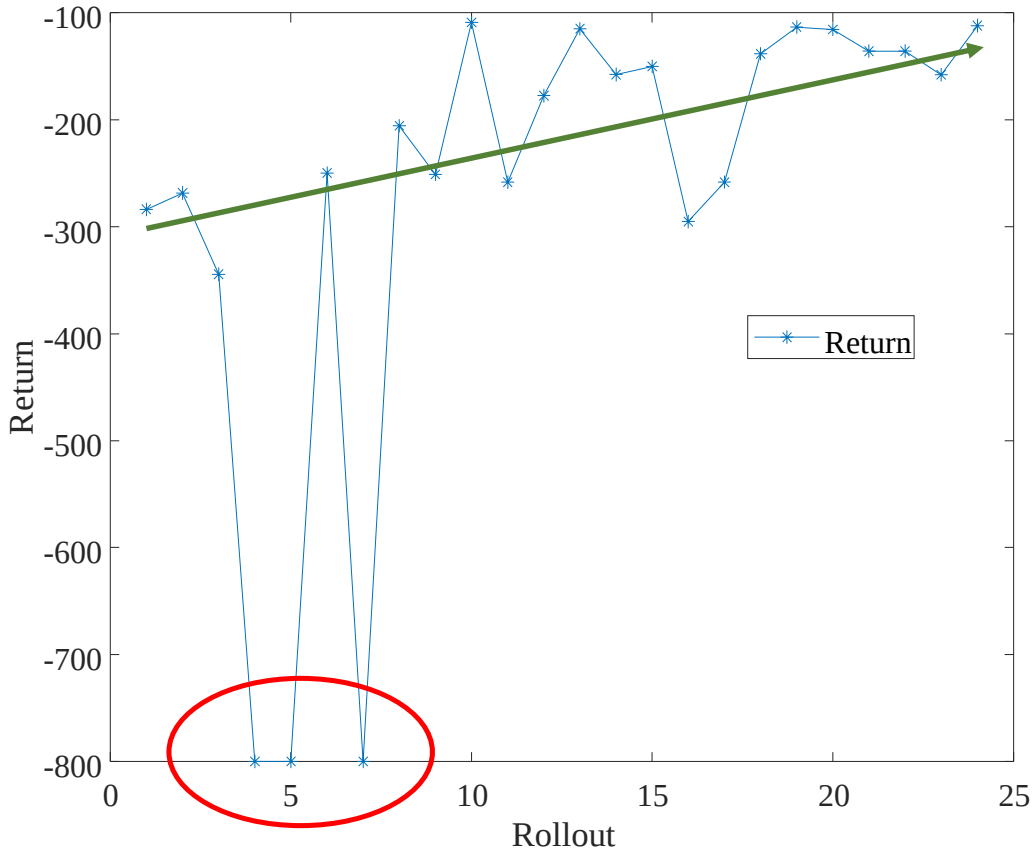
$$r_k = -w_{e_1}\left(v_k - v_{ref}{}_k\right)^2 - w_{e_2} I_k^2 - w_{e_3} U_k^2$$

- Policy parametrization:

$$U_k = \lambda_1\left(v_k - v_{ref}{}_k\right) + \lambda_2 \sum_{i=0}^{k}\left(v_{\mathrm{i}} - v_{ref}{}_{\mathrm{i}}\right) + \lambda_3 I_k + \lambda_4 \sum_{i=0}^{k} I_{\mathrm{i}} - \lambda_5 F_{h_k} \varsigma$$

Model-free design: proof-of-concept validation

- Return for 24 trials: repeated conditions, same user
- Enough to "learn" λ

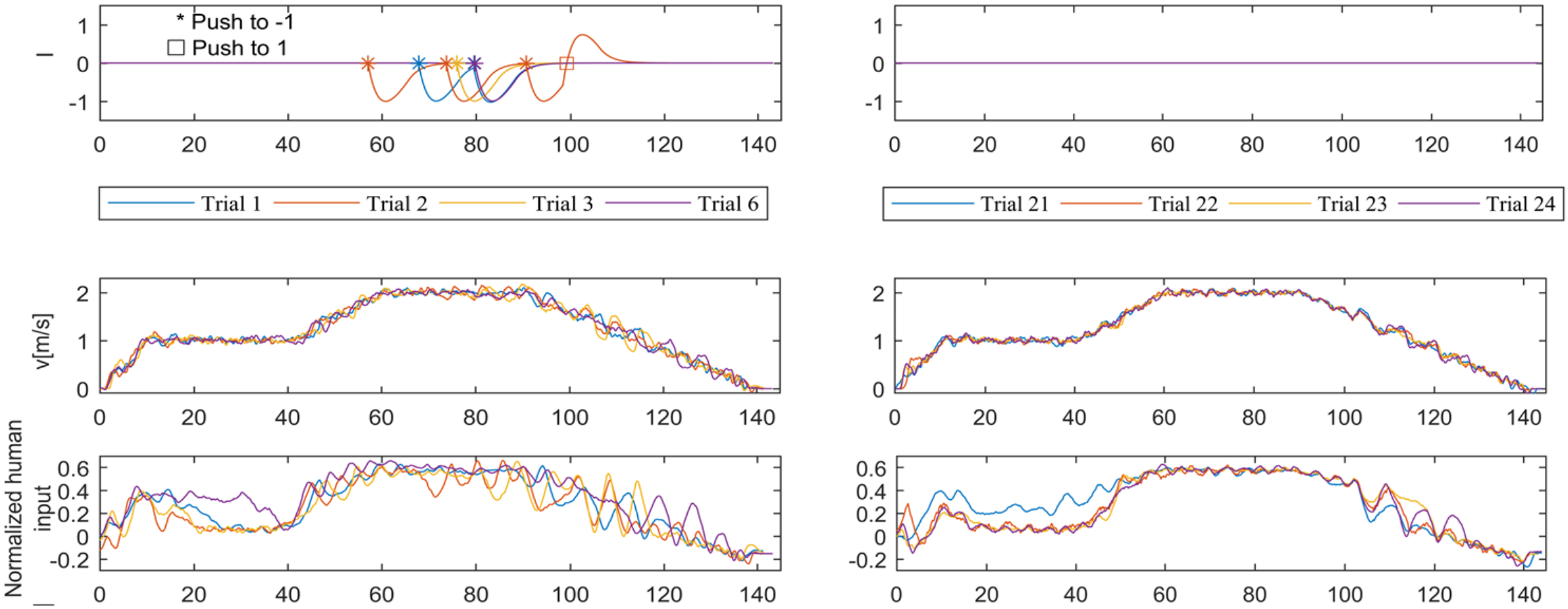


Unstable: very low return

Model-free design: proof-of-concept validation

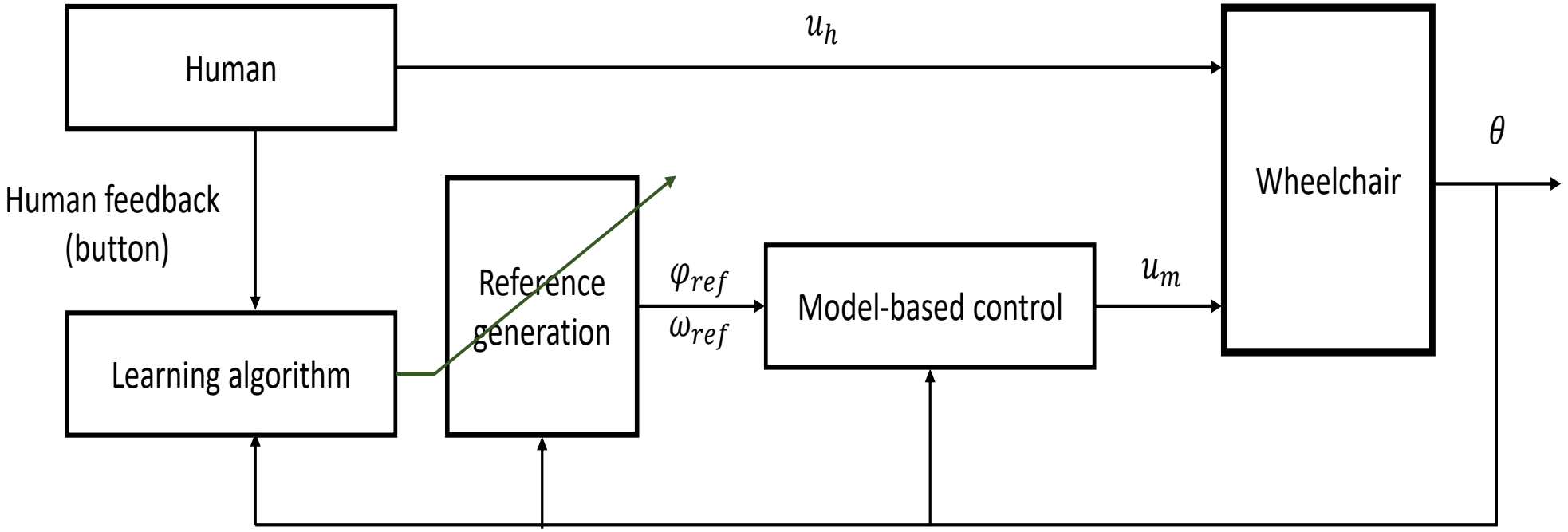
Beginning

After training



To go on

- Model-based control to track the trajectory desired by the user (low level control).
- Learning-control approach: learning the optimal parameters of the reference generation according to the user (high level control).



Thanks

Publications

International Journals

G. Feng, L. Buşoniu, T.M. Guerra, S. Mohammad (2019) – Data-Efficient Reinforcement Learning for Energy Optimization Under Human Fatigue Constraints of Power-Assisted Wheelchairs – IEEE Transactions on Industrial Electronics, Special Section on: Artificial Intelligence in Industrial System, 66 (12), 9734-9744 (IF 7.05)

International Conferences

Feng, G., Guerra, T. M., Nguyen A. T., Busoniu, L., & Mohammad, S. "Robust Observer-Based Tracking Control Design for Power-Assisted Wheelchairs". 5th IFAC Conference on Intelligent Control and Automation Sciences 21-23 August 2019, Belfast, Northern Ireland

Feng, G., Buşoniu, L., Guerra, T. M., & Mohammad, S. (2018, June). Reinforcement Learning for Energy Optimization Under Human Fatigue Constraints of Power-Assisted Wheelchairs. Annual American Control Conference (ACC) 27-29 June 2018 (pp. 4117-4122). IEEE.

Feng, G., Guerra, T. M., Mohammad, S., & Busoniu, L. "Observer-Based Assistive Control Design Under Time-Varying Sampling for Power-Assisted Wheelchairs". The 3rd IFAC Conference on Embedded Systems, Computational Intelligence and Telematics in Control June.6-8, 2018, Faro, Portugal IFAC-PapersOnLine, 51(10), 151-156.

Feng, G., Guerra, T. M., Busoniu, L., & Mohammad, S. "Unknown input observer in descriptor form via LMIs for power-assisted wheelchairs". In *2017 36th Chinese Control Conference (CCC)* (pp. 6299-6304). IEEE.

Workshop

Guerra, T. M., **Feng, G.**, Buşoniu, L., & Mohammad, S. "An example on trying to mix control and learning: power assisted wheelchair". *2nd Workshop Machine Learning Control (wMLC-2)*, Valenciennes, France, janvier 20.