Robotics and Autonomous Systems 197 (2026) 105293

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

# Passive Reinforcement Learning with Optimal Control for Safe Convergence in Cyber–physical Systems☆

Nicola Piccinelli [a,1,*], Daniele Meli [b,1], Enrico Bonoldi [a], Riccardo Muradore [a]

[a] Department of Engineering for Innovation Medicine, University of Verona, strada Le Grazie 15, Verona, 37135, Italy
[b] Department of Computer Science, University of Verona, strada Le Grazie 15, Verona, 37135, Italy

ARTICLE INFO

*Keywords:*
Reinforcement Learning
Robotics
Optimal control
Passivity based control
Safe learning
Real-world deployment

ABSTRACT

Reinforcement Learning (RL) can compute optimal strategies for accomplishing difficult tasks in complex scenarios. However, most RL algorithms do not provide safety and performance guarantees during the deployment phase. This is a critical drawback when RL is applied to cyber–physical systems such as robotic manipulators, where the goal is to always and safely converge to a desired goal or equilibrium state. Specifically, one fundamental safety requirement for robotic systems is closed-loop $\mathcal{L}_2$-stability, which has *passivity* as a sufficient condition. This paper proposes a novel switched RL control scheme for robotic systems, with *passivity and asymptotic stability guarantees*. This combines RL over constrained Markov decision processes, for *passive training and inference*, with Linear Quadratic Regulation (LQR) for asymptotic convergence to the desired equilibrium point. During RL training, the energy stored in the system is monitored via the virtual energy tank approach to train a cost critic function. During inference, the virtual energy tank modulates the command input to guarantee passivity. Finally, the reward design of the RL agent is based on the Lyapunov function associated with LQR control, in order to steer the system state towards the LQR basin of attraction, where a switching mechanism is triggered to guarantee asymptotic convergence. We compare our methodology with a model-based controller and other RL and model-based architectures applied to a paradigmatic under-actuated cart–pole system, an instance of a 2-DOF robotic manipulator, both in simulation and on a real setup. We also test the generality of our approach, with an experiment on a 6-DOF manipulator in simulation. The experimental validation shows that our methodology performs better in training and inference, even in the presence of plant modelling errors, while guaranteeing passivity and safety in the presence of large disruptive disturbances.

Reinforcement Learning (RL) is a well-established approach to optimal decision-making, which does not require any prior information about the plant and the environment. This is appealing for complex cyber–physical systems, e.g., robots [1]. However, safely transferring an RL strategy from simulation to real systems (sim-to-real) is a significant challenge [2] due to the lack of safety guarantees during policy exploration and the modelling gap with the uncertain world.

Safe RL aims to ensure that an agent learns while maintaining safety constraints throughout training and deployment [3]. One common approach is to impose constraints on the policy structure using prior domain knowledge. Another method involves designing backup policies, either through expert demonstrations or predefined safety mechanisms. Additionally, reward shaping can be used to guide the agent towards safe behaviour, though conflicting reward signals may reduce its effectiveness.

This paper considers two different aspects of safety for cyber–physical systems. First, from the perspective of dynamical systems, the safe deployment of an RL policy requires *asymptotic convergence*, i.e., the system must always converge to the specified goal (or equilibrium) state. Second, the RL policy shall not generate infeasible or disruptive commands, leading to system divergence. Then, we exploit *passivity* as a fundamental energy-related safety requirement for cyber–physical systems, as it ensures closed-loop $\mathcal{L}_2$ stability [4]. Specifically, we address the challenge of enforcing passivity in learning-based control schemes [5]. Passivity-Based Control (PBC), while effective for ensuring stability, often comes at the cost of reduced task performance (and possibly non-convergence) [6], a limitation that optimal learning-based strategies can mitigate. Previous works have explored different approaches to integrating passivity with learning-based control. For

☆ This article is part of a Special issue entitled: 'Planning & Learning' published in Robotics and Autonomous Systems.

* Corresponding author.
*E-mail address:* nicola.piccinelli@univr.it (N. Piccinelli).

[1] Equal contribution.

https://doi.org/10.1016/j.robot.2025.105293

Available online 19 December 2025

instance, [7] proposed using neural approximators to replace a PD controller for stabilising an inverted pendulum. However, their method is highly task-specific and does not generalise beyond the training conditions. More recently, [8,9] incorporated PBC into a reinforcement learning (RL) framework, ensuring passivity through episode pruning. However, they design a tank-based layer in the control scheme to enforce passivity at inference time (i.e., after training), thus only realising simple stability.

In contrast, we propose a novel RL-based methodology for the control of cyber–physical systems (e.g., robotic manipulators), realising asymptotic convergence to a desired equilibrium point under passivity constraints. Specifically, we define the problem as a Constrained Markov Decision Process (CMDP), with RL solving a constrained optimisation problem via Lagrangian relaxation [10]. The constraint function models a virtual energy tank [6], monitoring the physical energy flow between plant and controller. The energy tank approach to Passivity-Based Control (PBC) is domain-agnostic, hence being particularly appealing in combination with model-free control and optimisation architectures. The constrained RL agent aims to safely steer the system state in proximity of the desired equilibrium, dealing with the non-linearities of the plant. Then, to guarantee asymptotic convergence when close to the equilibrium, drawing inspiration from [11], we integrate RL with model-based optimal control for linear systems via Linear Quadratic Regulation (LQR) [12]. Unlike [11], we do not rely on LQR-based reward shaping, which may be ineffective in addressing convergence complexity [13] and lead to sub-optimal solutions [14]. Instead, we adopt a switched control scheme, leveraging a Lyapunov criterion defined over the linearisation of the cyber–physical system around the desired equilibrium. This formulation relaxes RL convergence requirements, simplifies the definition of a local Lyapunov function, and enhances the synergy between model-free and model-based optimal control, improving computational efficiency.

The main contributions of this paper are:

- modelling the passive RL problem as a CMDP, with tank-based cost. This fosters passive exploration during training, resulting in more stable convergence to the optimum and better performance than other RL architectures.
- theoretical proof that, endowing the RL-controlled plant with PBC guarantees passivity, the switched controller is able to provide the asymptotic stability of the overall system;
- empirical evaluation in a paradigmatic under-actuated cart–pole system, showing that our architecture performs better than other RL and model-based schemes. We also transfer the policy on a real system to prove the feasibility of our approach for real robotic applications, showing convergence under plant model uncertainty and non-divergence under exogenous disturbances. Finally, we tested our approach on a 6-DOF manipulator to show its scalability on more complex systems.

The paper is organised as follows: in Section 1, we present the related works to the proposed methodology and in Section 2 we provide the mathematical background. In Section 3, we describe the proposed methodology. In Section 4, we validate it on a simulation setup for the cart–pole system (Section 4.3), and we present the sim-to-real evaluation of the control architecture (Section 4.4). We also validate the generality of our methodology, by studying the stability of a simulated 6-DOF robotic manipulator (Section 4.5). Finally, in Section 5, we draw conclusions and outline possible future works.

## 1. Related works

Safe RL has gained much attention from the AI research community in the last decade [3]. In this context, two main approaches are available. The first methodology is to exploit available prior knowledge to define safety constraints on the policy structure [15,16]. This restricts the set of learnable policies, improving training efficiency and safety. However, it is often infeasible in practical complex domains due to the lack of sufficiently accurate mathematical models, harming generalisation. A different approach is to use task demonstrations gathered from experts or past system executions, either to learn a safe backup policy to be interleaved with the RL policy in inference [17], or to improve training efficiency [18], e.g., for dexterous robotic manipulation [19]. In this setting, the quality and generality of selected demonstrations are crucial, practically limiting application to complex safe and critical scenarios such as surgical robotics [20]. Another solution to safe RL is to shape the reward with additional signals pushing the agent towards desirable behaviour, e.g., with logical specifications defining risks and constraints [21,22]. Nonetheless, this converts the RL problem into a multi-objective optimisation, which is generally inefficient when evaluating many or contrasting reward signals [14]. The alternative approach considered in this paper is to model the RL environment as a CMDP [23], i.e., defining a cost signal similar to the classical reward but acting as a constraint to the RL optimisation problem. This can be solved, e.g., with Lagrangian optimisation [10]. As shown in autonomous driving [24], CMDPs provide better safety guarantees than reward shaping techniques. However, they have yet to be applied to the problem of passive (hence stable) and energy-aware control of cyber–physical systems.

Related to safety, another crucial requirement for cyber–physical systems is the *asymptotic convergence* to a desired equilibrium. When solving this problem with RL, typically, the equilibrium is embedded in the reward function as the task goal. Though the convergence of RL has been formally proved by [13], the rate of convergence depends on the approximation quality (e.g., the number of parameters) of the policy function. When RL is applied to a nonlinear control system, usually the aim is for the policy to converge to actions which realise *asymptotic stability* to a pre-defined goal state. Hence, while RL can easily achieve *simple equilibrium stability*, realising *asymptotic stability* is often much harder, depending on the specific convergence rate. Exploiting prior domain knowledge at the learning stage can improve the convergence rate towards the asymptotic equilibrium [25]. Recently, Lyapunov-based reward [26] and constraint shaping [27] have been proposed to address both the convergence and safety issues by exploiting Lyapunov theory from dynamical systems theory. However, RL is typically applied to non-linear and high-dimensional problems; hence, defining a suitable Lyapunov function constitutes another optimisation problem on its own [28].

Regarding Passivity-Based Control (PBC), a domain-agnostic approach to enforce passivity-by-design uses a virtual energy tank equipped with an initial energy budget [6]. The tank dynamics monitors the physical energy flow between the plant, the environment and the controller. The closed-loop system is passive and stable whenever the energy level is positive. When the energy is going to become negative, the control action is modulated to guarantee a positive energy level. Thus, PBC via a virtual energy tank is generally adopted when it is difficult to accurately model the system and derive sufficient stability conditions. The main drawback of PBC is the loss of performance due to the lack of optimisation during the control design [6]. Thus, the energy tank has been integrated into optimisation frameworks for shared autonomy and mobile robotics [29,30]. It is also particularly appealing when dealing with unknown or unbounded communication delay in bilateral teleoperation [31], combined with optimisation and MPC-based control [32,33]. Finally, to properly employ PBC, the energy tank needs to be initialised with a certain amount of energy (the so-called task energy); in [34], a constructive way of setting the initial energy in the tank based on the estimated energy consumption is provided.

## 2. Background

We now introduce relevant notation and background for the key blocks of our methodology, namely CMDP, optimal linear control via LQR, and the virtual energy tank approach to guarantee passivity.

### 2.1. Constrained Reinforcement Learning (C-RL)

RL assumes a decision-making problem is formalised as a MDP, represented by a tuple $\langle S, A, T, \rho, \gamma \rangle$. $S$ is the *state* space; $A$ is the set of *actions*; $\mathcal{T} : S \times A \to S$ is the *transition function* mapping the current state into the next state, given an action executed by the agent; $\rho : S \times A \to \mathbb{R}$ is the *reward function* providing a score for each action–state pair; $\gamma \in (0, 1]$ is a discount factor to penalise long sequences of actions. The goal of RL is to learn a policy map $\pi : S \to A$, which returns the best action at any given state, resulting in a (potentially infinite) sequence of state–action pairs $\tau = \langle s_0, a_0, s_1, a_1, \ldots \rangle$. We then say that the sequence $\tau$ follows the policy $\pi$, $\tau \sim \pi$. In a more practical setting with a finite time horizon $T$, actions are selected to maximise the *expected (discounted) return* or *value function*

$$J(\pi) = \mathbb{E}_{\tau \sim \pi} \left[ \sum_{i=0}^{T} \gamma^i \rho(s_i, a_i) \right] \tag{1}$$

In the popular actor-critic RL framework [35], $J(\pi)$ and $\pi$ are approximated by two (deep) neural networks, the critic and the actor, respectively. The actor network is modelled as $\pi_\theta = \pi(\cdot|s; \theta)$, where the parameters $\theta$ are updated (together with parameters of the critic network) iteratively as the agent gathers batches of experience from the environment, by exploiting a gradient-descent method $\nabla_\theta J(\pi_\theta)$. More formally, we recall the degree of approximation of a parametric model (e.g., a deep neural network) for the RL policy as follows.

**Definition 1** (*$\epsilon$-Universal Approximation [13]*)**.** A parametrisation $\pi_\theta$ is an $\epsilon$-universal approximation for $\pi$ if, for some $\epsilon > 0$, there exists a set of parameters $\bar{\theta}$ such that the following inequality holds

$$\max_{s \in S} \int_A \| \pi(a \mid s) - \pi_{\bar{\theta}}(a \mid s) \| \, da \leq \epsilon \tag{2}$$

The degree of approximation $\epsilon$ depends on the specific neural architecture, e.g., the number of parameters $\theta$.

In Constrained RL (C-RL), a CMDP is defined as $\langle S, A, T, \rho, \gamma, \{C_i\} \rangle$, where $C_i : S \times A \to \mathbb{R}$, $i = 1, \ldots, N$ are *costs* similar to negative rewards, associated with undesired state–action pairs for the agent. Hence, we can define $N$ *cost value functions* (i.e., one for each inequality constraint) in the form

$$J_{C_i}(\pi) = \mathbb{E}_{\tau \sim \pi} \left[ \sum_{i=0}^{\infty} \gamma^i C_i(s_i, a_i) \right] \tag{3}$$

and define the C-RL problem as maximising (1) while subject to $J_{C_i}(\pi) < d_i$, where $d_i \in \mathbb{R}$, $i = 1, \ldots, N$, is a threshold defining the relaxation of the cost constraint $C_i$. For ease of notation, from now on, we assume that costs (3) are written as a single cost function $C$ (with corresponding threshold vector $d$) and thus as a single cost value function $J_C(\pi)$. Hence, the actor-critic architecture can be extended to C-RL, considering an additional cost critic network $J_C(\pi_\theta)$ parametrised by $\theta$. To account for this constraint, we consider the specific C-RL formulation based on Lagrangian optimisation [10], and update $\theta$ according to the gradient $\nabla_\theta \mathcal{L}$, where $\mathcal{L}$ is the overall loss function defined as

$$\mathcal{L} = \frac{1}{1 + \lambda} \left( J(\pi_\theta) - \lambda J_C(\pi_\theta) \right) \tag{4}$$

and $\lambda$ is the Lagrange multiplier. At each batch iteration $k$, $\lambda$ is updated according to the following rule

$$\lambda_k \leftarrow (\lambda_{k-1} + K(\bar{J}_C(\pi_\theta) - d))_+ \tag{5}$$

where $K \in \mathbb{R}_+$, $\bar{J}_C(\pi_\theta)$ denotes the average of cost value estimations at batch $k$, and $(\cdot)_+$ ensures that $\lambda_k > 0$. In other words, the above update rule is a sort of integral control law[2] which increases $\lambda_k$ when the (average) cost constraint is violated, $\bar{J}_C(\pi_\theta) - d > 0$. When the constraint is not violated, $\lambda_k$ is set to zero, and the update of the actor depends only on the value critic $J(\pi_\theta)$.

The results in [13] establish the asymptotic convergence of both RL and Lagrangian C-RL. Specifically, for an $\epsilon$-universal approximation of the optimal policy in a Markov Decision Process (MDP) or a Constrained MDP (CMDP) with bounded rewards, the optimality gap is bounded by

$$\Delta = J(\pi^*) - J(\pi_\theta) \leq \frac{\epsilon}{1 - \gamma} \tag{6}$$

with $\gamma \in (0, 1)$. $J(\pi^*)$ represents the return of the optimal policy $\pi^*$, and $J(\pi_\theta)$ represents the return of the learned policy $\pi_\theta$. In the case of C-RL, this bound is achieved within a finite number of Lagrangian update steps [13].

Without loss of generality, in this paper, we consider the Deep Deterministic Policy Gradient (DDPG) architecture [36] for actor-critic RL, which supports continuous state and action spaces, hence being more realistic for its adoption on cyber–physical systems. However, any other actor-critic architecture may well be employed.

### 2.2. Virtual energy tank

Let

$$\boldsymbol{B}(\boldsymbol{q})\ddot{\boldsymbol{q}}(t) + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}(t) + \boldsymbol{D}\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{\tau}(t) + \boldsymbol{\tau}_e(t) \tag{7}$$

be the dynamic equations of motion of an $n$-DOF manipulator, where $\boldsymbol{q}(t) \in \mathbb{R}^n$ are the generalised coordinates. The joint torques $\boldsymbol{\tau}(t), \boldsymbol{\tau}_e(t) \in \mathbb{R}^n$ are the control and external torques, respectively. The matrices $\boldsymbol{B}(\boldsymbol{q}) \succ 0$, $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})$, $\boldsymbol{D} \succeq 0$ are the inertia, Coriolis and centrifugal, and friction terms, respectively [37]. The vector $\boldsymbol{g}(\boldsymbol{q}) = \frac{\partial V(\boldsymbol{q})}{\partial \boldsymbol{q}}$ represents the joint torques due to the potential energy. In this setting, we assume rigid links, and thus, we do not have any other sources of potential energy except the gravitational field [38].

Let $\mathcal{H}(\boldsymbol{q}(t))$ be the total energy of the manipulator

$$\mathcal{H}(\boldsymbol{q}(t)) = \frac{1}{2}\dot{\boldsymbol{q}}^T(t)\boldsymbol{B}(\boldsymbol{q})\dot{\boldsymbol{q}}(t) + V(\boldsymbol{q}(t)) \tag{8}$$

where $\frac{1}{2}\dot{\boldsymbol{q}}^T(t)\boldsymbol{B}(\boldsymbol{q})\dot{\boldsymbol{q}}(t)$ is the kinetic energy and $V(\boldsymbol{q}(t))$ is the potential energy. Considering the passivity condition for mechanical systems [38], the following power relationship holds

$$\dot{\mathcal{H}}(\boldsymbol{q}) \leq \dot{\boldsymbol{q}}^T(t)(\boldsymbol{\tau}(t) + \boldsymbol{\tau}_e(t)) \tag{9}$$

where the function $\mathcal{H}(\boldsymbol{q})$ can be seen as an energy storage function for the manipulator dynamics (7) [38]. This means that the overall energy stored in the system is always bounded by the already stored energy and the energy supplied at the power ports $(\boldsymbol{\tau}(t), \dot{\boldsymbol{q}}(t))$ and $(\boldsymbol{\tau}_e(t), \dot{\boldsymbol{q}}(t))$ [39]. Assuming the manipulator dynamics (7) passive [38] and given the power inequality (9), PBC aims at designing a controller making the closed-loop system with the manipulator passive at the environment power port $(\boldsymbol{\tau}_e(t), \dot{\boldsymbol{q}}(t))$, i.e.,

$$\dot{\mathcal{H}}(\boldsymbol{q}(t)) - \dot{\boldsymbol{q}}^T(t)\boldsymbol{\tau}(t) \leq \dot{\boldsymbol{q}}^T(t)\boldsymbol{\tau}_e(t), \tag{10}$$

via a virtual energy tank. Let the virtual energy tank dynamics be

$$\begin{cases} \dot{x}_t(t) = u_t(t) \\ y_t(t) = \frac{\partial E(x_t)}{\partial x_t} = x_t(t) \end{cases} \tag{11}$$

where $x_t(t) \in \mathbb{R}$ is the state of the tank and $(u_t(t), y_t(t)) \in \mathbb{R} \times \mathbb{R}$ is the power port through which the tank can exchange energy with the rest of the world, and let

$$E(x_t) = \frac{1}{2}x_t^2(t) \tag{12}$$

be the energy stored in the tank [32]. The robot (7) is interconnected to the energy tank and uses the energy stored in the tank to implement the control action $\boldsymbol{\tau}(t)$. This can be done by implementing the following

[2] A more sophisticated control law could be used as in [10]. However, this did not introduce significant benefits in our experiments.

power-preserving interconnection between the robot and the energy tank

$$\begin{cases} \boldsymbol{\tau}(t) = \boldsymbol{\omega}(t)\, y_t(t) \\ u_t(t) = -\boldsymbol{\omega}^T(t)\dot{\boldsymbol{q}}(t) \end{cases} \tag{13}$$

where $\boldsymbol{\omega}(t) \in \mathbb{R}^n$ is the modulation factor. By substituting (13) in (7), we get the manipulator dynamics coupled with the energy tank and controlled through $\boldsymbol{\omega}(t)$

$$\boldsymbol{B}(\boldsymbol{q})\ddot{\boldsymbol{q}}(t) + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}(t) + \boldsymbol{D}\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{\omega}(t)x_t(t) + \boldsymbol{\tau}_e(t). \tag{14}$$

It is worth mentioning that, thanks to the power interconnection, the power used by the manipulator's actuators $\dot{\boldsymbol{q}}^T(t)\boldsymbol{\tau}(t)$ reduces the available energy in the tank.

By monitoring the energy $E(x_t)$, we can modulate the desired control action $\bar{\boldsymbol{\tau}}(t)$ computed by the controller to prevent the energy tank depletion. This may happen, e.g., in the case of a non-passive controller (as in the generic RL case) or in the presence of delays. The modulation factor $\boldsymbol{\omega}(t)$ is

$$\boldsymbol{\omega}(t) = \begin{cases} \bar{\boldsymbol{\tau}}(t)x_t^{-1}, & \text{if } E(x_t) \geq \epsilon \\ 0, & \text{otherwise} \end{cases} \tag{15}$$

where $\epsilon > 0$ is a positive number to avoid singularities in (11). We do not explicitly consider saturation on the upper bound $E^{\max}$ to keep the notation simple. This can be done using a switching variable limiting energy storage, as shown in [40].

### 2.3. Linear Quadratic Regulation

The LQR [12] is an optimal control technique for LTI dynamical systems in the form

$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} \\ \boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D} \end{cases} \tag{16}$$

with the aim of minimising the quadratic cost function

$$J = \int_0^\infty \left(\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{u}^T\boldsymbol{R}\boldsymbol{u}\right) dt$$

with $\boldsymbol{Q} \succ 0$ (positive definite matrix) and $\boldsymbol{R} \succ 0$ (positive definite matrix). If $(\boldsymbol{A}, \boldsymbol{B})$ is stabilisable and $(\boldsymbol{A}, \boldsymbol{Q}^{1/2})$ is detectable, then the optimal steady-state feedback control law is given by $\boldsymbol{u} = -\boldsymbol{K}\boldsymbol{x}$, where

$$\boldsymbol{K} = \boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P} \tag{17}$$

is the LQR gain obtained from the unique solution $\boldsymbol{P} \succ 0$ of the Algebraic Riccati Equation (ARE) [4]

$$\boldsymbol{A}^T\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P} + \boldsymbol{Q} = \boldsymbol{0}. \tag{18}$$

The positive definite quadratic Lyapunov function

$$V(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{P}\boldsymbol{x}, \tag{19}$$

satisfies $\dot{V}(\boldsymbol{x}) < 0$ along trajectories of the closed-loop system. This condition ensures that the system is globally asymptotically stable in the equilibrium point $\boldsymbol{x} = \boldsymbol{0}$.

## 3. Methodology

We now present our methodology for the safe asymptotic convergence of cyberphysical systems. The overall architecture during the inference phase is sketched in Fig. 1. The core idea of our methodology is to linearise the model of the robotic system in (7) around a desired equilibrium point (without loss of generality, $\boldsymbol{q} = \boldsymbol{0}, \dot{\boldsymbol{q}} = \boldsymbol{0}$), resulting in (16) by setting $\boldsymbol{x} = \left[\boldsymbol{q}^T\dot{\boldsymbol{q}}^T\right]^T$. Then, we design a hybrid controller, switching between *Passive Reinforcement Learning (P-RL)*, i.e., C-RL with an energy tank for passive stability and a local LQR to stabilise the system around the desired equilibrium point. As mentioned in Section 2.3, LQR stabilisation is proved with the Lyapunov function $V(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{P}\boldsymbol{x}$. We exploit such information in two ways:

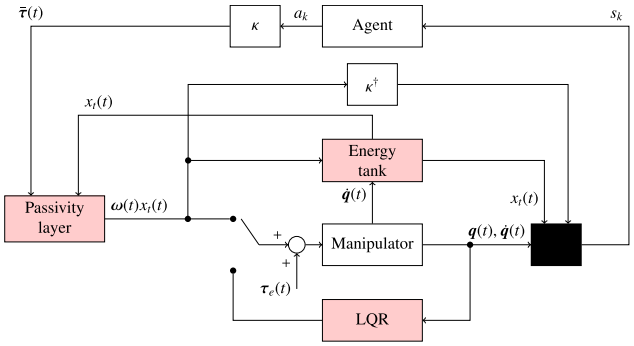



**Fig. 1.** Full control architecture during inference. During deployment, the passivity layer ensures passivity, while the switching mechanism facilitates convergence to equilibrium once the cart–pole enters the basin of attraction $\boldsymbol{\Omega}$.

- in P-RL, we shape the reward of the agent based on $V(\boldsymbol{x}) \leq \sigma$, where $\sigma \in \mathbb{R}_+$ is a suitably small constant which defines the extent of the *basin of attraction* where the LQR can guarantee asymptotic stability according to Lyapunov indirect criterion (see Section 3.1) [41]. In this way, the RL agent is responsible for bringing the system into the basin of attraction. We remark that differently unlike other approaches [26,27,42], *we do not require an accurate model of the real non-linear system*, since $V(\boldsymbol{x})$ is defined for the linear system;
- in the switching between P-RL and LQR, we perform local regulation only when the linear approximation around the equilibrium is sufficiently accurate (hence, $V(\boldsymbol{x}) \leq \sigma$).

### 3.1. LQR basin of attraction

One key part of our methodology is the definition of the *basin of attraction*, which is a sufficiently small region around the equilibrium where the Lyapunov indirect criterion guarantees that the asymptotic stability of $\boldsymbol{x} = 0$ for the linearised system with LQR control implies the asymptotic stability of the non-linear system with the same control. In particular, we define the basin starting from the Lyapunov function of the linearised system around the equilibrium. More specifically, let

$$\Omega(\sigma) = \left\{\boldsymbol{x} \;\middle|\; \|\boldsymbol{x}\|_P^2 \leq \sigma\right\}, \quad \|\boldsymbol{x}\|_P := \sqrt{\boldsymbol{x}^\top P\boldsymbol{x}}, \tag{20}$$

be the sublevel set of $V(\boldsymbol{x}) = \boldsymbol{x}^\top P\boldsymbol{x}$. For a suitable $\sigma \in \mathbb{R}_+$, the linear approximation of the system in $\Omega(\sigma)$ is sufficiently accurate to guarantee the asymptotic stability of the controlled non-linear system. Then, the set $\Omega(\sigma)$ also provides our *switching criterion*: once $\boldsymbol{x}$ enters it, we switch to the local LQR controller and achieve asymptotic stability to the desired equilibrium.

To this aim, let the time derivative of the Lyapunov candidate function $\dot{V}(\boldsymbol{x})$ *along the trajectory of the non-linear system dynamics*, represented as

$$\dot{\boldsymbol{x}} = (\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K})\,\boldsymbol{x} + \boldsymbol{f}_{\text{nl}}(\boldsymbol{x}), \tag{21}$$

where $\boldsymbol{A}_{\text{cl}} = \boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}$ is the linear part (from LQR design around the equilibrium), and $\boldsymbol{f}_{\text{nl}}(\boldsymbol{x})$ collects the non-linear terms vanishing at the equilibrium, $\boldsymbol{f}_{\text{nl}}(\boldsymbol{0}) = \boldsymbol{0}$. Using (18) and (19) one obtains

$$\dot{V}(\boldsymbol{x}) = \boldsymbol{x}^\top\boldsymbol{M}\,\boldsymbol{x} \;+\; 2\,\boldsymbol{x}^\top\boldsymbol{P}\,\boldsymbol{f}_{\text{nl}}(\boldsymbol{x}), \tag{22}$$

with $\boldsymbol{M} = \boldsymbol{A}_{\text{cl}}^\top\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A}_{\text{cl}}$ in the LQR case.

The objective is to identify the largest sublevel set $\Omega(\sigma)$ that remains within the basin of attraction of the closed-loop nonlinear system. Since this set cannot be computed analytically, we introduce a Monte Carlo procedure based on randomised boundary tests. For $\sigma > 0$, let

$$S_\sigma = \{\boldsymbol{x} \in \mathbb{R}^n : \|\boldsymbol{x}\|_P^2 = \sigma\} \tag{23}$$

denote the boundary ellipsoid. To generate samples uniformly on $S_\sigma$, we draw $\boldsymbol{v} \sim \mathrm{Unif}(\mathbb{S}^{n-1})$ on the Euclidean unit sphere, yielding

$$\boldsymbol{x} = \sqrt{\sigma}\, P^{-1/2} \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|_2} \tag{24}$$

which guarantees $\|\boldsymbol{x}\|_P^2 = \sigma$. This induces the surface measure $\mu_{P,\sigma}$ on $S_\sigma$ used for boundary sampling. We require that

$$\dot{V}(\boldsymbol{x}) \le -\gamma, \quad \forall\, \boldsymbol{x} \in S_\sigma, \tag{25}$$

for some margin $\gamma \ge 0$. Since (25) cannot be verified exhaustively, we approximate it by sampling $N$ i.i.d. points $\boldsymbol{x}_s \sim \mu_{P,\sigma}$ via (24). Each test produces Bernoulli outcomes

$$Z_s(\sigma) = \begin{cases} 1 & \dot{V}(\boldsymbol{x}_s) \le -\gamma, \\ 0 & \text{otherwise} \end{cases} \qquad s = 1, \dots, N. \tag{26}$$

The number of successful checks is then a binomial random variable

$$S(\sigma) = \sum_{s=1}^{N} Z_s(\sigma) \;\sim\; \mathrm{Binomial}(N, p(\sigma)) \tag{27}$$

with success probability

$$p(\sigma) = \mu_{P,\sigma}\big(\{\boldsymbol{x} \in S_\sigma : \dot{V}(\boldsymbol{x}) \le -\gamma\}\big). \tag{28}$$

We say that the boundary test *accepts* $\sigma$ if $S(\sigma) = N$ (no violations detected). To locate the largest admissible level, we embed the randomised test into a bisection search over $\sigma$ in logarithmic coordinates. Let $I_0 = [\log \sigma_{\min}, \log \sigma_{\max}]$ be an initial interval with a guaranteed feasible point at $\sigma_{\min}$ and an infeasible point at $\sigma_{\max}$. At iteration $k$, we define

$$\sigma_k = \exp\Big(\tfrac{1}{2}(L_k + R_k)\Big) \tag{29}$$

where $[L_k, R_k]$ is the current log-interval. The boundary test at $\sigma_k$ induces the update

$$(L_{k+1}, R_{k+1}) = \begin{cases} (\log \sigma_k,\ R_k), & \text{if accept,} \\ (L_k,\ \log \sigma_k), & \text{if reject.} \end{cases} \tag{30}$$

with $L_0 = \log \sigma_{min}$ and $R_0 = \log \sigma_{max}$ After $K$ iterations, the maximal feasible radius is found along that stochastic search path

$$\hat{\sigma}^\star = \exp(L_K). \tag{31}$$

Since each boundary test is random, the outcome of every bisection step is random as well; consequently, $\hat{\sigma}^\star$ itself is a random variable

$$\hat{\sigma}^\star \in \mathbb{R}_+, \quad \hat{\sigma}^\star \sim \nu \tag{32}$$

for some induced distribution $\nu$ determined by the dynamics, the Lyapunov function, and the sample size $N$. Repeating the full bisection with independent seeds yields a collection $\{\hat{\sigma}_i^\star\}_{i=1}^M$ of i.i.d. samples from the distribution of $\hat{\sigma}^\star$. From this empirical distribution, we choose an empirical distribution and select a deterministic conservative threshold

$$\sigma = \rho\, \phi\big(\{\hat{\sigma}_i^\star\}_{i=1}^M\big), \quad \rho \in (0, 1], \tag{33}$$

where $\phi$ is a statistical functional (e.g., the empirical mean, a lower quantile, or the minimum over the sample set).

### 3.2. P-RL training

Enforcing passivity during training encourages the development of passive policies within a given energy budget. To achieve this, we introduce the Passive Reinforcement Learning (P-RL) framework, built upon C-RL and Lagrangian optimisation, as shown in Fig. 2. In particular, we integrate a virtual energy tank into the C-RL structure to monitor the energy flow through the controller's power port $(\boldsymbol{\tau}(t), \dot{\boldsymbol{q}}(t))$. This virtual energy tank for PBC operates independently of the system and control architecture, making it well-suited for black-box controllers

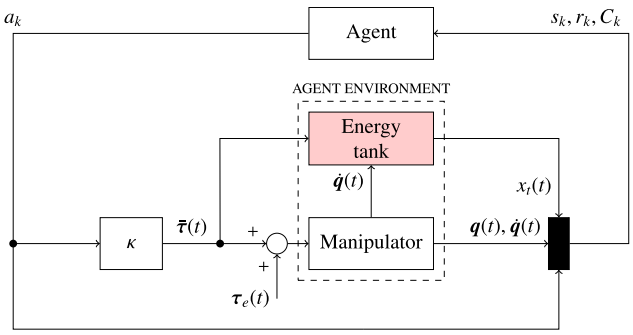



**Fig. 2.** Passive RL architecture for training. During the training phase, the agent's environment does not include the switching mechanism, and the passivity layer remains inactive to allow the constraints to be enforced during learning.

such as deep neural networks. Below, we outline the training process where the RL problem is formulated as a CMDP.

The reward of the agent is defined as

$$R = -\max(0, V(\boldsymbol{x}) - \sigma). \tag{34}$$

In this way, we encourage the agent to approach the basin of attraction of the LQR controller, corresponding to $\Omega$ in (20). The constraint of the CMDP is defined as

$$C_k = -(x_{t,k} - \sqrt{2\varepsilon}) \tag{35}$$

and at each time step $k$, we want $C_k \le 0$, i.e., $x_{t,k} \ge \sqrt{2\varepsilon}$, in order to preserve passivity. The notation $(\cdot)_k$ represents the discrete time step $k$ such that $(\cdot)_k = (\cdot)(kT_s)$, where $T_s$ is the sample time.

Then the cost value function $J_C(\pi)$ in (3) estimates the tank dynamics $\hat{x}_{t,k}$ due to the action $\boldsymbol{\tau}$ selected by the RL agent to maximise a generic task-specific value function (1). From the definition of the tank energy in (12), the passivity condition in (15) becomes $\hat{x}_t \ge \sqrt{2\varepsilon}$. Thus, requiring the passivity of the RL agent is equivalent to imposing the constraint $J_C(\pi) \le d$ within the optimisation problem, where $d = -\sqrt{2\varepsilon}$. The resulting CMDP problem can be solved via Lagrangian optimisation (4). At this stage, estimating with sufficient accuracy $E^{max}$ and $E^{ini}$, namely, the initial tank level, is crucial. Starting from an arbitrarily high level may result in never breaking the condition $E(x_t) \le \varepsilon$, even though the real system is practically unstable since it cannot consume so much energy in the simulation horizon. In Section 4, we propose to estimate $E^{max}$ and $E^{ini}$ (set to the same value) starting from the energy required by a state-of-the-art optimal controller to complete the task.

### 3.3. P-RL inference and switched controller

The designed P-RL agent acts as a force/torque control for a cyber–physical system modelled by the dynamics defined in (7). The learned policy is such that $\bar{\boldsymbol{\tau}}_k = \kappa(a_k)$, where $a_k = \pi_{\theta_k}$ and $\kappa : A \to \mathbb{R}^n$ is the time-invariant map projecting the RL action space to the manipulator control set and $\kappa^\dagger : \mathbb{R}^n \to A$ goes in the opposite direction.

When employing the RL-based controller, we want the interconnection between the agent and manipulator to be passive, also *during inference*, to guarantee the closed-loop passivity and thus keep the system $\mathcal{L}_2$ stable. In fact, the C-RL formulation still *does not guarantee* the passivity of the agent during inference for two reasons. Lagrangian optimisation solves the dual problem of the CMDP, which does not guarantee by itself that the primal problem is also feasible. Moreover, C-RL may only learn the passivity cost in the limited state space the agent explores during training. Hence, a trained policy is not guaranteed to be passive by itself *in every possible state*. On the contrary, the passivity is guaranteed only by the closed-loop dynamics (14). This

becomes even more crucial when deploying an RL policy trained in simulation to the real world. In fact, training assumes an ideal model of the underlying cyber–physical system, whose inaccuracy may break the passivity constraint. As a consequence, we still have to monitor the tank level and enforce passivity by modulating the desired torque $\bar{\boldsymbol{\tau}}(t)$, using the passivity layer in Fig. 1 with control modulation (15).

The overall control system is thus a switched system composed of the two controllers such that

$$\boldsymbol{\tau}_k = \begin{cases} -\boldsymbol{K}\boldsymbol{x}_k, & V(\boldsymbol{x}) \leq \sigma \\ \boldsymbol{\omega}_k, & \text{otherwise} \end{cases} \tag{36}$$

where the switching condition is based on (20). The stability of the switched controller will be discussed in the following.

### 3.4. Theoretical results

We now want to prove two fundamental theoretical results related to our methodology:

- the *passivity (hence simple stability) guarantees* of our P-RL architecture during inference (Fig. 1). This is a crucial result since modelling inaccuracies during simulated training inevitably affect the P-RL agent. Hence, it is important to guarantee that its behaviour will not diverge to ensure safety.
- the *asymptotic convergence to the equilibrium* of the proposed switched P-RL and LQR architecture, thanks to the definition of the basin of attraction and the switching control law explained in Section 3.1.

**Proposition 1.** *The learned policy $\pi_\theta$ for the CMDP problem makes the architecture in Fig. 1 passive with respect to the pair $(\boldsymbol{\tau}_e(t), \dot{\boldsymbol{q}}(t))$.*

**Proof.** Consider as a storage function the total energy (at time $t$) of the manipulator endowed with the energy tank as defined in (11)

$$W(t) = \mathcal{H}(t) + E(x_t(t)) \tag{37}$$

where $\mathcal{H}$ according to (8) is the stored energy in the manipulator. It is possible to write the power of the system by taking the derivative of (37) with respect to time such that

$$\dot{E}(x_t(t)) = u_t(t)y_t(t) = -\boldsymbol{\omega}(t)\dot{\boldsymbol{q}}^T(t)y_t(t) = -\dot{\boldsymbol{q}}^T(t)\kappa(\pi_\theta(t)) \tag{38}$$

with $\pi_\theta(t) = \pi_{\theta_k}, t \in [kTs, (k+1)Ts]$. Considering the power balance (9)

$$\dot{W}(t) \leq \dot{\boldsymbol{q}}^T(t)\kappa(\pi_\theta(t)) - \dot{\boldsymbol{q}}^T(t)\kappa(\pi_\theta(t)) + \dot{\boldsymbol{q}}^T(t)\boldsymbol{\tau}_e(t) \tag{39}$$

hence

$$\int_0^t \dot{W}(\tau)d\tau \leq \int_0^t \dot{\boldsymbol{q}}^T(s)\boldsymbol{\tau}_e(s)ds \tag{40}$$

and thus

$$W(t) - W(0) \leq \int_0^t \dot{\boldsymbol{q}}^T(s)\boldsymbol{\tau}_e(s)ds \tag{41}$$

which corresponds to the passivity condition of the P-RL controller. □

Using PBC with discrete-time systems may introduce inaccuracies in energy monitoring and potentially lead to a loss of passivity. However, as in [8], by assuming constant forces in the sampling time, (41) is equivalent to

$$W((k+1)T_s) - W(kT_s) \leq \int_{kT_s}^{(k+1)T_s} \dot{\boldsymbol{q}}^T(s)\boldsymbol{\tau}_e(s)ds. \tag{42}$$

The asymptotic convergence of our overall architecture is proved by the following theorem.

**Theorem 1.** *The switching control architecture, with the basin of attraction defined by the set $\Omega$ according to (20), makes the closed-loop system asymptotically stable to the equilibrium $\boldsymbol{x} = \boldsymbol{0}$.*

**Proof.** Let $\boldsymbol{x} \in \mathbb{R}^n$ be the state vector evolving under the control law (36). Let us use as a candidate Lyapunov function the following quadratic form

$$V(\boldsymbol{x}) = \boldsymbol{x}^T\boldsymbol{P}\boldsymbol{x}, \quad \boldsymbol{P} = \boldsymbol{P}^T \succ 0 \tag{43}$$

where $\boldsymbol{P}$ is obtained from the solution of the ARE (18). We aim to show that such a function is a global Lyapunov function, thus ensuring the asymptotic stability of the closed-loop system. We proceed by analysing the closed-loop system dynamics when $\boldsymbol{x} \in \Omega$. In this case, we have

$$\dot{\boldsymbol{x}} = (A - BK)\boldsymbol{x} \tag{44}$$

and since the LQR gain is computed such that $(A - BK)$ is Hurwitz, we have

$$\dot{V}(\boldsymbol{x}) = -\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} < 0, \quad \forall \boldsymbol{x} \neq 0 \tag{45}$$

and thus, under LQR, $V(\boldsymbol{x})$ is strictly decreasing, ensuring asymptotically stability.

Let us now consider the case when $\boldsymbol{x} \notin \Omega$. According to the reward map (34) and the optimality gap condition (6), the optimal P-RL action is passive and eventually satisfies

$$V(\boldsymbol{x}) < \sigma \tag{46}$$

i.e., $\exists T$ s.t. $V(\boldsymbol{x}(t)) < \sigma \ \forall t > T$. Thus, when the system state falls outside $\Omega$, the RL policy drives the system back to the basin of attraction $\Omega$. Hence, the system cannot remain in any level set of $V(\boldsymbol{x})$, except at $\boldsymbol{x} = 0$. Therefore, the trajectory of $\boldsymbol{x}$ converges asymptotically to the equilibrium. □

## 4. Experimental results

The proposed methodology has been validated using two cyber–physical systems: a cart–pole and an UR5 serial manipulator. The cart–pole scenario with a 2D inverted pendulum has been tested both in simulation using Matlab 2024b and on a real setup. The serial manipulator scenario, on the other hand, has been tested in simulation only to show the feasibility of the proposed approach on a larger and more complex system. We performed the training phase on an Intel i7-8700 using an NVIDIA Quadro P4000, and the inference phase for the cart–pole was performed on an Intel i5-1135G7. On the cart–pole system, we compare our C-RL methodology against two benchmarks:

**RL** the agent uses the same reward function (34), but neglecting the tank state $x_{t,k}$ and not including the passivity constraint during learning;

**ES** the local LQR controller is combined with a model-based energy-based swing-up method [42].

In the following, we will first evaluate the performance at training and inference time of the cart–pole, both in simulation and on the real system. Then we will show the adaptation of the proposed method to work with a simulated 6-DOF serial manipulator.

### 4.1. The cart–pole system

The cart–pole scenario represents a 2-DOF instance of a generic robotic manipulator described by (14), with only one actuated joint at the cart, see Fig. 3. Let

$$\boldsymbol{q} = \begin{bmatrix} x \\ \alpha \end{bmatrix} \tag{47}$$

be the vector of the generalised coordinates (pole angle and cart displacement), $\boldsymbol{q} \in \mathcal{Q} \subset \mathbb{R}^2$, and let $F \in \mathcal{U} \subset \mathbb{R}$ be the linear force applied on the cart. The force is converted to the control input as

$$\bar{\boldsymbol{\tau}}_k = \kappa(a_k) = \begin{bmatrix} F & 0 \end{bmatrix}^T. \tag{48}$$

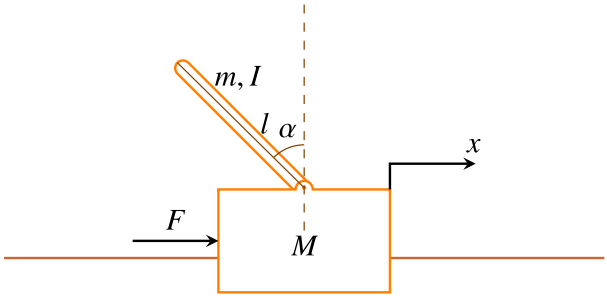



**Fig. 3.** The cart–pole scenario used to validate the proposed passive RL controller. The command force is $F$, the mass of the cart is $M$, the mass of the pendulum is $m$ and $I$ denotes its inertia, $l$ is the length of the pole, $\alpha$ is the pole angle, and $x$ is the cart displacement.

The mechanical parameters of our real setup in Fig. 6 are the following. The viscous friction of the pole and the cart are estimated to be $F_p^v = 0.01\,\mathrm{N\,s\,rad^{-1}}$ and $F_c^v = 5\,\mathrm{N\,s\,m^{-1}}$, respectively, while the mass of the cart and the pole are 150 g and 80 g. The pendulum length is 21.5 cm; we assume constant density to easily derive the centre of mass and the moment of inertia. These mechanical parameters are used to linearise the cart–pole system for designing the LQR controller.

The cart can move on a linear guide with maximum length $x^{\max}$, and the force $F$ is limited such that

$$\mathcal{U} = \{F \mid -F^{\max} \leq F \leq F^{\max}\} \tag{49}$$

with $F^{\max} = 15\,\mathrm{N}$. We formulate our problem as a CMDP, where $A = \{a_k \in \mathcal{U}\}$ is the set of possible actions $a_k$ and $S$ is the set of states $s_k$ at time $k$

$$s_k = \begin{bmatrix}\sin(\alpha_k) & \cos(\alpha_k) & \dot{\alpha}_k & x_k & \dot{x}_k & x_{t,k}\end{bmatrix}^T. \tag{50}$$

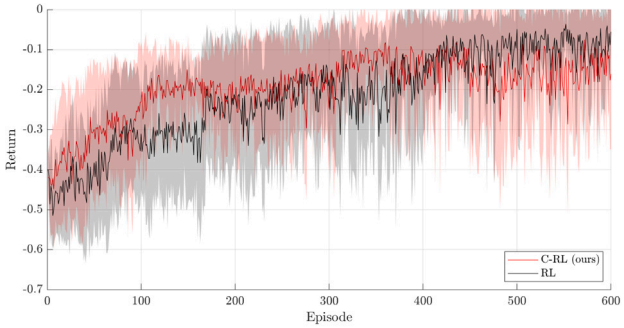
### 4.2. Training results

During training in simulation, we replicated the physical constraints of our real setup. Hence, the episodes were pruned if the cart went outside the linear guide length of the real setup

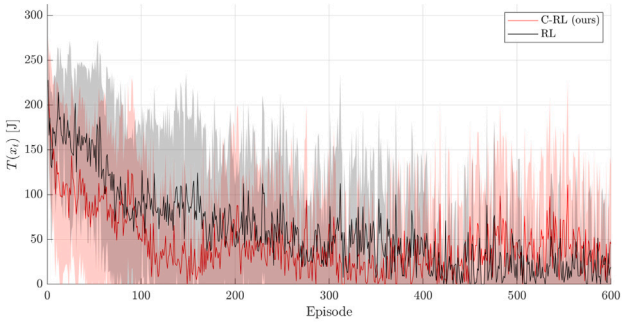
$$|x_k| - \frac{1}{2}x_{\max} > 0, \tag{51}$$

with $x_{\max} = 40\,\mathrm{cm}$ for our experimental setup. It is worth highlighting that the proposed scheme does not implement an early stop due to tank depletion to let the agent learn the cost critic properly. The energy tank needs to be initialised with a certain amount of energy to employ PBC properly. We followed a constructive way of setting the initial energy in the tank based on the estimated energy consumption [34]. So, we defined the episode length and the tank parameters $E^{\max}$ and $E^{\mathrm{ini}}$ based on the energy consumption of the ES model-based control scheme. The value for the initialisation of the energy tank resulted in $x_t = 25$, leading to an initial energy budget of 312 J. The learning rate for the actor and the critics was set to 0.001. We trained all the agents for 600 episodes over random seeds and used the average reward per step as a more representative metric to select the best agent. During training, we remark that the switching to LQR was not enabled.

Fig. 4 shows the evolution of the training curve across episodes, comparing RL (green) vs. C-RL (red). In Fig. 4(a), the average reward per step, computed as the cumulative reward divided by the steps taken in the episode, shows that C-RL converges slightly faster to the optimal policy than RL, thanks to tank observation helping in the initial exploration phase. Fig. 5(d) shows no significant difference between the energy depleted by RL and C-RL during training. However, in the next sections, we will show the crucial advantages of our methodology at inference time, where energy awareness results in better and more efficient task completion.




(a) Average reward per step




(b) Residual energy in the tank

**Fig. 4.** Training results over the episodes (mean ± std. dev.) for the C-RL (in red) and RL (in grey) policies. (a) the average cumulative reward per step, (b) the remaining amount of energy $T(x_t)$ at the end of each episode. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

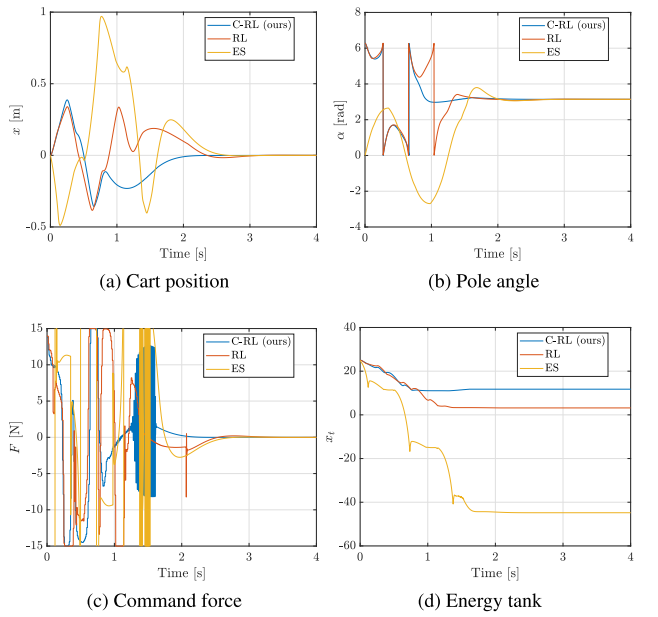
### 4.3. Simulation results

We evaluated the best agents in inference using RL and C-RL and the ES baseline in simulation to assess the performance at stabilising the cart–pole at $\boldsymbol{x} = \mathbf{0}$. Here, we introduced the switching mechanism to the local LQR controller. As shown in Fig. 5, all methodologies were able to stabilise the system. However, the advantage of C-RL vs. ES is evident in Figs. 5(a)–5(c)–5(d). The cart motion range, shown in Fig. 5(a), was consistently within the maximum range $x_{\max} = 40$ cm when the RL and C-RL controllers were used. This is not the case for the ES controller, where the maximum displacement of the cart reached almost 1 metre of distance from the centre.

Fig. 5(c) shows the overall command force profile to the cart. Our C-RL methodology showed command peak forces only at the beginning of the task (within 1 s from the start), while RL and ES apply higher (potentially dangerous) force commands more persistently. This is also evident from the evolution of the energy tank state variable $x_t$, shown in Fig. 5(d), where the C-RL architecture was the one accomplishing the task with the least amount of energy. This demonstrated the effect of training the agent using the passivity constraint and having the energy tank level in the observation. It is worth mentioning that the energy tank is monitoring the energy only if the reinforcement learning controller is active, since its PBC action is not required when the cart pole is inside the basin of attraction, i.e., under the LQR control.

### 4.4. Real setup results

To verify the effectiveness of the proposed methodology, we evaluated C-RL, RL and ES architectures with the switching criterion proposed in Section 3.3 in a real setup. The experimental setup, shown in Fig. 6, was composed of a custom-made inverted pendulum system




(a) Cart position (b) Pole angle

(c) Command force (d) Energy tank

**Fig. 5.** A comparison of the system evolution in simulation with the best trained C-RL (in blue), RL (in orange) policies and the ES controller (in yellow). (a) position of the cart; (b) angular position of the pole; (c) command force of the switched controller; (d) energy tank state $x_t$ (the plot did not show the actual energy $\frac{1}{2}x_t^2$ since it highlights the tank depletion). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

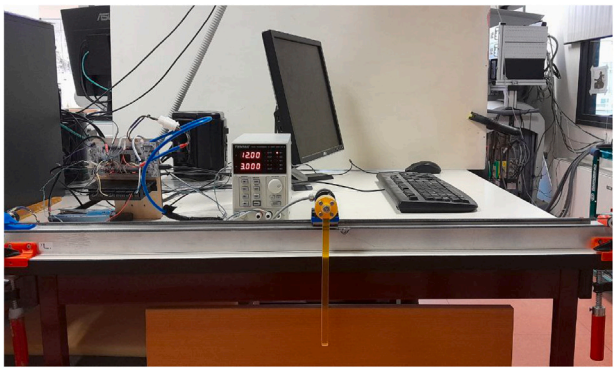
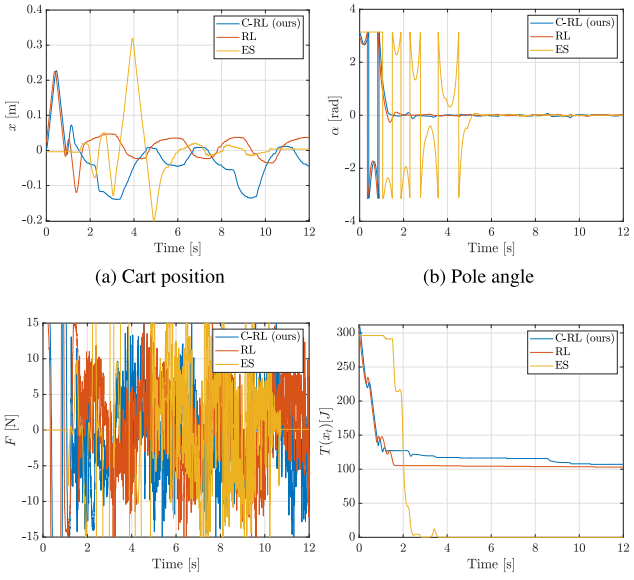


**Fig. 6.** The cart–pole experimental setup. The system is composed of a linear guide and a 3D-printed pole. The system is controlled using a brushed DC motor interfaced with an STM microcontroller.

controlled via ROS 2 (Robot Operating System). The motor was a brushed DC model controlled via a current loop. It is equipped with a relative encoder and a gearbox, while the pole angle was measured using an absolute encoder. The electronics consist of a microcontroller running a Zephyr-based firmware and interfaced with ROS 2 through Ethernet communication. A detailed description of the hardware is available in Table 1. The host machine was running two ROS nodes: the first one updating the current cart–pole system state and then sending the desired force command to the microcontroller (bridge node) through the socket connection. The second one, implementing the switched control scheme (controller node). We performed two different experiments: (i) *stabilisation* at $\boldsymbol{x} = \mathbf{0}$, to assess the robustness to the uncertain estimation of physical parameters in a real cyber–physical system *sim-to-real gap*; (ii) *disturbance rejection*, where the pole is suddenly moved away from the equilibrium configuration. During the experimental evaluation, we used $F^{\max} = 20\,\mathrm{N}$.

**Table 1**
Hardware components and software environment for the low-level controller used for the experimental setup.

| Component | Description |
|---|---|
| Motor | Pololu HP #4842 9.68:1 gearbox |
| Motor controller | Maxon Motors ESCON 50/5 #409510 |
| Cart encoder | Quadrature 48 CPR |
| Pole encoder | Baumer BMMH 30 SSI |
| Microcontroller | STM32 Nucleo-F767ZI |
| Firmware | Zephyr RTOS-based firmware |



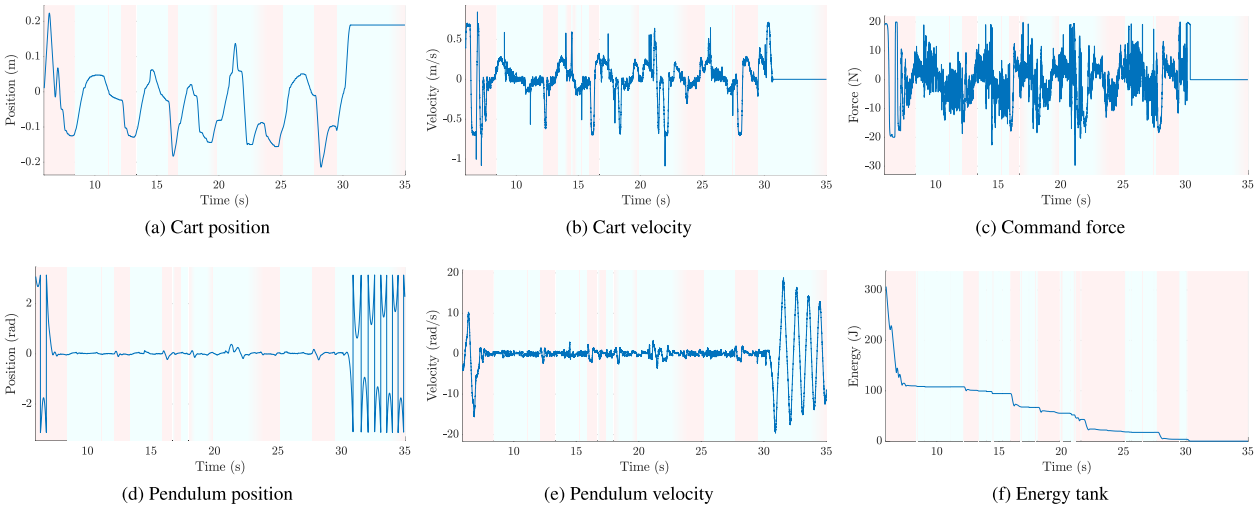

(a) Cart position (b) Pole angle

(c) Command force (d) Energy tank

**Fig. 7.** Stabilisation results on the real setup controlled with the best trained C-RL (in blue), RL (in orange) policies and ES controller (in yellow). (a) position of the cart; (b) angular position of the pole; (c) command force of the switched controller; (d) the energy level in the tank $T(x_t) = \frac{1}{2}x_t^2$. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

In Fig. 7, we report the stabilisation results. As a first key result, the RL policy and the C-RL policy were both able to stabilise the real cart–pole system. Indeed, for both, the cart displacement did not converge to 0 (Fig. 7(a)) while the pendulum converged to the desired configuration in a similar way (Fig. 7(b)). Moreover, as shown in Fig. 7(a), the cart position displacement induced by C-RL (blue) was similar to RL, about 20 cm. Similarly to the simulation results, the ES controller (yellow) was able to stabilise the cart–pole, but with a larger motion. Moreover, from Fig. 7(b), C-RL stabilises much faster (less than 3 s) with respect to ES (5 s). Fig. 7(d) shows that all architectures eventually consumed the whole amount of available energy. However, we remark that our C-RL architecture is the only one guaranteeing passivity (hence safe non-divergence) according to the passivity layer in Fig. 1 and Proposition 1. This will be evidenced especially in the following disturbance rejection tests. Finally, Fig. 7(c) shows that all architectures applied noisy command forces to the cart. This is due to the inaccuracy in the estimation of the real physical parameters (in particular, the stiction coefficient), hence the sim-to-real gap.

We now report a detailed analysis of the C-RL behaviour in the real environment in Fig. 8, where the pendulum was perturbed inside the estimated basin of attraction $\Omega$. Pink and green shaded areas represent the regions where the C-RL and LQR operate, respectively. At about 7 s the controller switched from C-RL to LQR, since the Lyapunov function (Fig. 9) went below the threshold $\sigma$. Once the LQR took over, the system remained in the basin of attraction. In between 12 s to 30 s, an external




(a) Cart position (b) Cart velocity (c) Command force

(d) Pendulum position (e) Pendulum velocity (f) Energy tank

**Fig. 8.** Disturbance rejection results on the real setup of the best learned C-RL policy on the real cart–pole system, the red and green shaded areas represent the action of the C-RL agent and the LQR, respectively. (a,b) position and velocity of the cart; (c) command force to the cart using the switched controller; (d,e) the position and velocity of the pendulum; (f) stored energy in the tank $T(x_t) = \frac{1}{2}x_t^2$. Pink and green shaded regions represent the regions where the C-RL and LQR operate, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

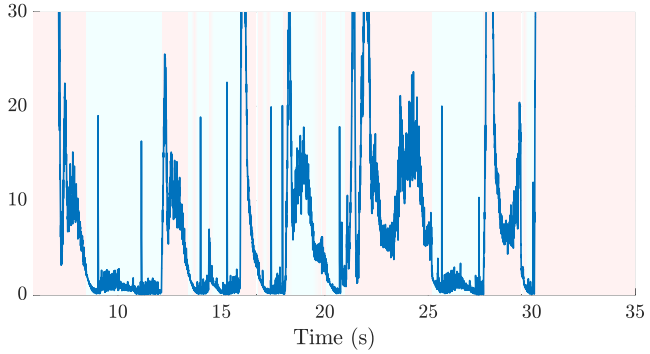
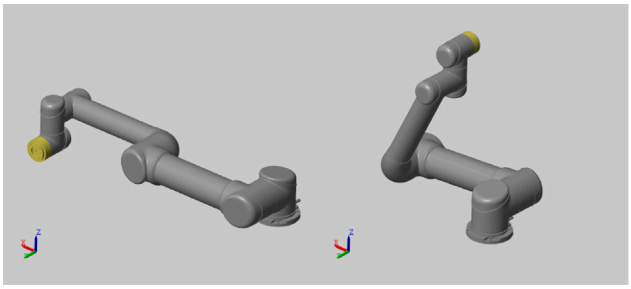



**Fig. 9.** The evolution of the Lyapunov function during the experimental evaluation on the real setup of the best C-RL policy. Pink and green shaded regions represent the regions where the C-RL and LQR operate, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)



(a) Initial configuration (b) Final configuration

**Fig. 10.** A visualisation of the initial and final configuration we used for testing the proposed methodology in the manipulator scenario.

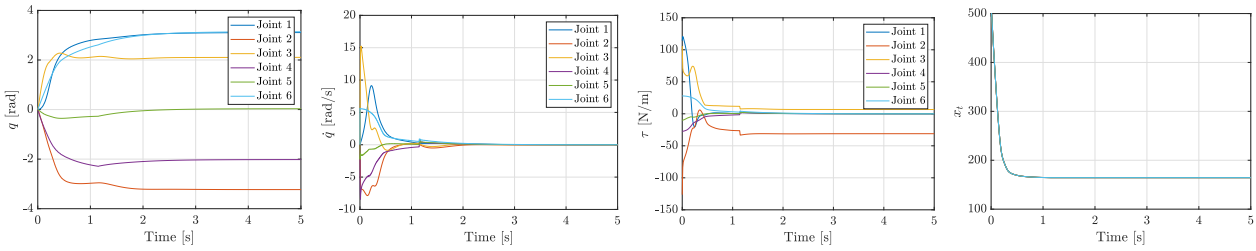
disturbance was applied to the pendulum, pushing it away from the equilibrium multiple times. The state of the system remained in $\Omega$, until a new disturbance came in action both from an external human interaction or inaccuracies in the real model, at 12 s, 16 s, 20 s, 25 s and 30 s. Thus, the LQR control was able to keep the cart position (Fig. 8(a)) and pole angle (Fig. 8(d)) stable. This behaviour was visible in Fig. 9, where every time the Lyapunov function grew above $\sigma$ caused the system to switch back to the C-RL policy. However, at 30 s the system received a large disturbance pushing the energy tank depletion (Fig. 8(f)), which due to the passivity layer in Fig. 1 set the command force to zero (Fig. 8(c)), hence the cart and pole velocities (Figs. 8(b)–8(e)), letting the system evolve in free motion without converging to the equilibrium anymore. This analysis proved the key importance of the passivity layer (see Proposition 1): even in the presence of modelling errors on the real cyber–physical system, our control architecture was able to reject small disturbances which keep the state in the basin of attraction, thanks to the asymptotic convergence guarantees of the LQR actions. When the disturbance grew too much, the system still remained safe, thanks to the simple stability guaranteed by PBC. In such a situation, by feeding back an energy budget to the controller, the C-RL would stabilise the system again.

### 4.5. The serial manipulator system

The serial manipulator model we used was a 6-DOF fully actuated UR5 robot, with $\boldsymbol{q} \in \mathcal{Q} \subset \mathbb{R}^n$ and $\dot{\boldsymbol{q}} \in \mathbb{R}^n$, with $n = 6$, being the joint positions and velocities, respectively. The robot was controlled via joint torques $\boldsymbol{\tau} \in \mathcal{U} \subset \mathbb{R}^n$. Similarly to the cart–pole, we linearised the manipulator's dynamic model, using the same methodology adopted in [43], around an equilibrium configuration $\boldsymbol{q}_t$ for designing the LQR controller. The initial and final configurations are shown in Fig. 10. We enforced the joint torque limits according to the UR datasheet, and we defined, as before, $A = \{a_k \in \mathcal{U}\}$ and the set of states $S$ at time $k$ as

$$s_k = \left[\boldsymbol{q}^T \dot{\boldsymbol{q}}^T x_{t,k}\right]^T. \tag{52}$$

As for the cart–pole, the episodes were pruned if the robot reached the joint position limits. In this scenario, the energy budget was computed by running a simple PD controller for reaching the desired configuration. The initial budget for the energy tank has been set to $x_t = 500$, while the learning rate for the actor and critics was set to 0.0005. We trained the agent for 2000 episodes, and as before, we used the average reward per step to select the best agent. In Fig. 11, we reported the behaviour of the best-trained policy using the proposed method applied to the serial manipulator. The transition between the C-RL and the LQR happened at 1.5 s, and as shown in Fig. 11(c), the command torques after the transition converged to the equilibrium torques necessary to compensate for the gravity. In Fig. 11(a), the evolution of the joint




(a) Joint position (b) Joint velocity (c) Command torque (d) Energy tank

**Fig. 11.** Stabilisation of a 6-DOF manipulator in simulation with the best trained C-RL. (a) joint positions; (b) joint velocities; (c) command torques of the switched controller; (d) energy tank state $x_t$.

positions over time is shown, and together with Fig. 11(b) shows the overall stabilisation of the system once it reached the desired equilibrium configuration. Finally, in Fig. 11(d), the evolution of the energy tank is shown. The policy has been trained with the energy constraint activated, and the learned policy successfully accomplishes the task without tank depletion.

## 5. Conclusions

In this paper, we proposed a novel switched control architecture that integrates passive reinforcement learning with optimal control to ensure safe convergence in cyber–physical systems and robotic manipulators. Our approach employs a constrained Markov Decision Process (CMDP) equipped with a virtual energy tank to enforce passivity during the learning phase and guarantee simple stability (hence safe non-divergence) at inference time. A Lyapunov-based switching mechanism is incorporated to guarantee asymptotic stability through Linear Quadratic Regulation (LQR). Crucially, our methodology requires minimal plant knowledge based solely on the Lyapunov function defined for the linearised system at the equilibrium and the model-free tank and C-RL approaches. We demonstrated the effectiveness of our method on an underactuated cart–pole system, an instance of a 2-DOF manipulator, both in simulation and on a real setup. We also tested the generality of our approach, studying the stability of a simulated 6-DOF robotic manipulator. We compared against a state-of-the-art model-based controller based on ES and RL with no tank. Our approach performs better in simulation tests, with minimal cart displacement, energy consumption and faster convergence. In the real experiments, the tank observation and constraint allow C-RL to overcome the modelling inaccuracies of the plant due to the inevitable sim-to-real gap. Moreover, the use of the energy tank during inference guarantees simple stability when large external disturbances are applied to the system, where the local LQR controller can only compensate for small perturbations.

Future work will extend this framework to more complex robotic tasks and investigate alternative optimal control strategies to further enhance safety and performance. For instance, these include a predictive tank refilling strategy to relax the initial assumption of the energy budget for the task [44]. Also, we will explore better control strategies to limit possible command discontinuities induced by our switched control system.

## CRediT authorship contribution statement

**Nicola Piccinelli:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Daniele Meli:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Formal analysis, Conceptualization. **Enrico Bonoldi:** Validation, Software, Data curation. **Riccardo Muradore:** Writing – review & editing, Supervision, Resources, Project administration, Formal analysis, Conceptualization.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.robot.2025.105293.

## Data availability

Data will be made available on request.

## References

[1] N. Akalin, A. Loutfi, Reinforcement learning approaches in social robotics, Sensors 21 (4) (2021) 1292.
[2] W. Zhao, J.P. Queralta, T. Westerlund, Sim-to-real transfer in deep reinforcement learning for robotics: a survey, in: 2020 IEEE Symposium Series on Computational Intelligence, SSCI, IEEE, 2020, pp. 737–744.
[3] J. Garcıa, F. Fernández, A comprehensive survey on safe reinforcement learning, J. Mach. Learn. Res. (2015).
[4] A. Van der Schaft, L2-Gain and Passivity Techniques in Nonlinear Control, Springer, 2000.
[5] S. Arimoto, S. Kawamura, F. Miyazaki, Convergence, stability and robustness of learning control schemes for robot manipulators, in: Proceedings of the International Symposium on Robot Manipulators on Recent Trends in Robotics: Modeling, Control and Education, 1986.
[6] F. Dimeas, N. Aspragathos, Online stability in human-robot cooperation with admittance control, IEEE Trans. Haptics 9 (2) (2016) 267–278.
[7] S. Massaroli, M. Poli, F. Califano, J. Park, A. Yamashita, H. Asama, Optimal energy shaping via neural approximators, SIAM J. Appl. Dyn. Syst. 21 (3) (2022) 2126–2147.
[8] R. Zanella, G. Palli, S. Stramigioli, F. Califano, Learning passive policies with virtual energy tanks in robotics, IET Control Theory Appl. (2024).
[9] R. Zanella, F. Califano, C. Secchi, S. Stramigioli, Learning passive policies, in: European Robotics Forum, Springer, 2024, pp. 338–343.
[10] A. Stooke, J. Achiam, P. Abbeel, Responsive safety in reinforcement learning by pid lagrangian methods, in: International Conference on Machine Learning, PMLR, 2020, pp. 9133–9143.
[11] S. Zoboli, V. Andrieu, D. Astolfi, G. Casadei, J.S. Dibangoye, M. Nadri, Reinforcement learning policies with local LQR guarantees for nonlinear discrete-time systems, in: 2021 60th IEEE Conference on Decision and Control, CDC, IEEE, 2021, pp. 2258–2263.
[12] B.D. Anderson, J.B. Moore, Optimal Control: Linear Quadratic Methods, Courier Corporation, 2007.
[13] S. Paternain, L. Chamon, M. Calvo-Fullana, A. Ribeiro, Constrained reinforcement learning has zero duality gap, Adv. Neural Inf. Process. Syst. 32 (2019).
[14] C.F. Hayes, R. Rădulescu, E. Bargiacchi, J. Källström, M. Macfarlane, M. Reymond, T. Verstraeten, L.M. Zintgraf, R. Dazeley, F. Heintz, et al., A practical guide to multi-objective reinforcement learning and planning, Auton. Agents Multi-Agent Syst. 36 (2022).
[15] Y. Du, J.-q. Li, X.-l. Chen, P.-y. Duan, Q.-k. Pan, Knowledge-based reinforcement learning and estimation of distribution algorithm for flexible job shop scheduling problem, IEEE Trans. Emerg. Top. Comput. Intell. (2022).

[16] S. Munikoti, D. Agarwal, L. Das, M. Halappanavar, B. Natarajan, Challenges and opportunities in deep reinforcement learning with graph neural networks: A comprehensive review of algorithms and applications, IEEE Trans. Neural Networks Learn. Syst. (2023).
[17] B. Thananjeyan, A. Balakrishna, S. Nair, M. Luo, K. Srinivasan, M. Hwang, J.E. Gonzalez, J. Ibarz, C. Finn, K. Goldberg, Recovery rl: Safe reinforcement learning with learned recovery zones, IEEE Robotics Autom. Lett. 6 (3) (2021) 4915–4922.
[18] H. Chen, C. Liu, Safe and sample-efficient reinforcement learning for clustered dynamic environments, IEEE Control Syst. Lett. 6 (2021) 1928–1933.
[19] S. Christen, S. Stevšić, O. Hilliges, Guided deep reinforcement learning of control policies for dexterous human-robot interaction, in: 2019 International Conference on Robotics and Automation, ICRA, IEEE, 2019, pp. 2161–2167.
[20] A. Pore, E. Tagliabue, M. Piccinelli, D. Dall'Alba, A. Casals, P. Fiorini, Learning from demonstrations for autonomous soft-tissue retraction, in: 2021 International Symposium on Medical Robotics, ISMR, IEEE, 2021, pp. 1–7.
[21] G. Mazzi, D. Meli, A. Castellini, A. Farinelli, Learning logic specifications for soft policy guidance in POMCP, in: Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems, 2023, pp. 373–381.
[22] D. Meli, A. Castellini, A. Farinelli, Learning logic specifications for policy guidance in POMDPs: an inductive logic programming approach, J. Artificial Intelligence Res. 79 (2024).
[23] A. Wachi, Y. Sui, Safe reinforcement learning in constrained Markov decision processes, in: International Conference on Machine Learning, PMLR, 2020, pp. 9797–9806.
[24] D. Kamran, T.D. Simão, Q. Yang, C.T. Ponnambalam, J. Fischer, M.T. Spaan, M. Lauer, A modern perspective on safe automated driving for different traffic dynamics using constrained reinforcement learning, in: 2022 IEEE 25th International Conference on Intelligent Transportation Systems, ITSC, IEEE, 2022, pp. 4017–4023.
[25] R. Kamalapurkar, P. Walters, W.E. Dixon, Model-based reinforcement learning for approximate optimal regulation, Automatica 64 (2016) 94–104.
[26] T. Westenbroek, F. Castaneda, A. Agrawal, S. Sastry, K. Sreenath, Lyapunov design for robust and efficient robotic reinforcement learning, in: Conference on Robot Learning, PMLR, 2023, pp. 2125–2135.
[27] Y. Chow, O. Nachum, E. Duenez-Guzman, M. Ghavamzadeh, A Lyapunov-based approach to safe reinforcement learning, Adv. Neural Inf. Process. Syst. 31 (2018).
[28] T.A. Johansen, Computation of Lyapunov functions for smooth nonlinear systems using convex optimization, Automatica 36 (11) (2000) 1617–1626.
[29] B. Capelli, C. Secchi, L. Sabattini, Passivity and control barrier functions: optimizing the use of energy, IEEE Robotics Autom. Lett. 7 (2) (2022) 1356–1363.
[30] F. Benzi, F. Ferraguti, G. Riggio, C. Secchi, An energy-based control architecture for shared autonomy, IEEE Trans. Robotics 38 (6) (2022).
[31] F. Loschi, N. Piccinelli, D. Dall'Alba, R. Muradore, P. Fiorini, C. Secchi, An optimized two-layer approach for efficient and robustly stable bilateral teleoperation, in: 2021 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2021, pp. 12449–12455.
[32] N. Piccinelli, R. Muradore, A bilateral teleoperation with interaction force constraint in unknown environment using non linear model predictive control, Eur. J. Control 62 (2021) 185–191.
[33] N. Piccinelli, R. Muradore, Linearized virtual energy tank for passivity-based bilateral teleoperation using linear MPC, IEEE Trans. Robotics (2025).
[34] C. Schindlbeck, S. Haddadin, Unified passivity-based cartesian force/impedance control for rigid and flexible joint robots via task-energy tanks, in: 2015 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2015, pp. 440–447.
[35] I. Grondman, L. Busoniu, G.A. Lopes, R. Babuska, A survey of actor-critic reinforcement learning: Standard and natural policy gradients, IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.) 42 (6) (2012) 1291–1307.
[36] T.P. Lillicrap, J.J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, D. Wierstra, Continuous control with deep reinforcement learning, 2015, arXiv preprint arXiv:1509.02971.
[37] B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo, Robotics: Modelling, Planning and Control, first ed., Springer Publishing Company, Incorporated, 2008.
[38] B. Brogliato, R. Lozano, B. Maschke, O. Egeland, et al., Dissipative systems analysis and control, Theory Appl. 2 (2007) 2–5.
[39] S. Stramigioli, C. Secchi, A.J. van der Schaft, C. Fantuzzi, Sampled data systems passivity and discrete Port-Hamiltonian systems, IEEE Trans. Robotics 21 (4) (2005) 574–587.
[40] F. Ferraguti, N. Preda, A. Manurung, M. Bonfe, O. Lambercy, R. Gassert, R. Muradore, P. Fiorini, C. Secchi, An energy tank-based interactive control architecture for autonomous and teleoperated robotic surgery, IEEE Trans. Robotics 31 (5) (2015).
[41] H. Khalil, Control Systems: An Introduction, Michigan Publishing Services, 2023.
[42] K. Furuta, M. Yamakita, S. Kobayashi, Swing up control of inverted pendulum, in: Proceedings IECON'91: 1991 International Conference on Industrial Electronics, Control and Instrumentation, IEEE, 1991, pp. 2193–2198.
[43] N. Piccinelli, R. Muradore, Interaction force constraints for position-controlled manipulator using linear MPC, in: 2023 21st International Conference on Advanced Robotics, ICAR, IEEE, 2023, pp. 176–182.
[44] D. Sacerdoti, F. Benzi, C. Secchi, A reinforcement learning-based control strategy for robust interaction of robotic systems with uncertain environments, in: 2024 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2024, pp. 5788–5794.