applied sciences

MDPI

Review

Motion Planning of Upper-Limb Exoskeleton Robots: A Review

Clautilde Nguiadem [1,2,3,*], Maxime Raison [1,2] and Sofiane Achiche [1]

[1] Department of Mechanical Engineering, École Polytechnique de Montréal, Montréal, QC H3T 1J4, Canada; maxime.raison@polymtl.ca (M.R.); sofiane.achiche@polymtl.ca (S.A.)
[2] Technopole in Pediatric Rehabilitation of Ste-Justine UHC, Montréal, QC H3T 1C5, Canada
[3] CRME—Research Center, Office GR-123, 5200, East Bélanger Street, Montréal, QC H1T 1C9, Canada
* Correspondence: clautilde.nguiadem@polymtl.ca; Tel.: +1-(514)-374-1710-8189

Received: 4 October 2020; Accepted: 26 October 2020; Published: 29 October 2020

Abstract: (1) Background: Motion planning is an important part of exoskeleton control that improves the wearer's safety and comfort. However, its usage introduces the problem of trajectory planning. The objective of trajectory planning is to generate the reference input for the motion-control system. This review explores the methods of trajectory planning for exoskeleton control. In order to reduce the number of surveyed papers, this review focuses on the upper limbs, which require refined three-dimensional motion planning. (2) Methods: A systematic search covering the last 20 years was conducted in Ei Compendex, Inspect-IET, Web of Science, PubMed, ProQuest, and Science-Direct. The search strategy was to use and combine terms "trajectory planning", "upper limb", and "exoskeleton" as high-level keywords. "Trajectory planning" and "motion planning" were also combined with the following keywords: "rehabilitation", "humanlike motion", "upper extremity", "inverse kinematic", and "learning machine ". (3) Results: A total of 67 relevant papers were discovered. Results were then classified into two main categories of methods to plan trajectory: (i) Approaches based on Cartesian motion planning, and inverse kinematics using polynomial-interpolation or optimization-based methods such as minimum-jerk, minimum-torque-change, and inertia-like models; and (ii) approaches based on "learning by demonstration" using machine-learning techniques such as supervised learning based on neural networks, and learning methods based on hidden Markov models, Gaussian mixture models, and dynamic motion primitives. (4) Conclusions: Various methods have been proposed to plan the trajectories for upper-limb exoskeleton robots, but most of them plan the trajectory offline. The review approach is general and could be extended to lower limbs. Trajectory planning has the advantage of extending the applicability of therapy robots to home usage (assistive exoskeletons); it also makes it possible to mitigate the shortages of medical caregivers and therapists, and therapy costs. In this paper, we also discuss challenges associated with trajectory planning: kinematic redundancy and incompatibility, and the trajectory-optimization problem. Commonly, methods based on the computation of swivel angles and other methods rely on the relationship (e.g., coordinated or synergistic) between the degrees of freedom used to resolve kinematic redundancy for exoskeletons. Moreover, two general solutions, namely, the self-tracing configuration of the joint axis and the alignment-free configuration of the joint axis, which add the appropriate number of extra degrees of freedom to the mechanism, were employed to improve the kinematic incompatibility between human and exoskeleton. Future work will focus on online trajectory planning and optimal control. This will be done because very few online methods were found in the scope of this study.

Keywords: automatic systems; trajectory planning; exoskeleton; imitation learning; inverse kinematics; machine learning

1. Introduction

Developments in assistive robotic devices offer enormous prospects in medical fields, such as surgical, rehabilitation, and assistive robotics [1]. Many robotic devices, referred to as exoskeletons, have been developed for patient rehabilitation and physical assistance. Their efficacy has been proved in these fields. For instance, rehabilitation robots interact with humans and can be successfully employed in the early phase of recovery from stroke [2–5], but despite great progress, this introduces the trajectory-planning problem, which is fundamental and widely dealt with in robotics. Indeed, in any autonomous robotic system, trajectory planning is of crucial importance in moving the robot from its initial position to the desired position because it finds the desired trajectory linking the two positions or configurations (start and goal). In robotic systems such as exoskeleton robots, the motion planner generates the desired trajectory, and the control algorithm ensures that the robot tracks this planned trajectory (see Figure 1). Consequently, trajectory planning is an essential tool for understanding and fine-tuning exoskeleton control. Because of the close physical interaction between exoskeletons and humans, the problem of planning suitable trajectories is more relevant for their usage in order to ensure user safety and comfort. Initially, exoskeletons were applied in rehabilitation for training rather than to help therapists to perform training on patients. Even if their effect was satisfactory, this method required the presence of the therapist [3,6]. Moreover, this approach restricted applications to clinical use. Currently, when using exoskeleton robots to assist rehabilitation therapy, the motion trajectory for training the user′s limb movements first needs to be planned. With an appropriate trajectory-planning method, exoskeletons can operate in realistic environments, such as those of daily life [7]. Accordingly, proper trajectory planning is essential to the use of exoskeleton robots in home settings. Much work was done on trajectory planning for exoskeleton robots [8–15]. In this context, this paper deals with trajectory-planning problems, i.e., the computation of desired motion profiles for the actuation system of automatic machines. We explore methods of trajectory planning used in the literature for upper-limb exoskeleton robots. In robotics, trajectory planning can either be done in task space or in configuration space for exoskeletons [16]. When trajectory planning is done in task space, inverse kinematics is required to obtain joint motion. However, it is difficult to solve the inverse kinematics problem because there is an infinite number of solutions [17]. Attempts to resolve the inverse kinematics problem in robotics have used many approaches, including Jacobian [12], optimization, dimensionality-reduction, and learning-based methods [18]. In [19], a method based on optimization was proposed to implicitly resolve inverse kinematics. In addition, since exoskeleton control is done at the joint level, trajectory planning in joint space makes it possible to avoid the calculation of inverse kinematics. Most planning methods in joint space rely on learning human behavior and the transfer of human skills to the robot through demonstration. However, demonstrations in joint space are strongly time-dependent [16]. Although methods/algorithms based on dynamic time warping (DTW) exist to deal with the time-dependence problem (as in [14]), their accuracy is low when applied to approximate high dimensions. In addition to the above challenges, environmental changes related to activities-of-daily-life (ADL) task scenarios add difficulties in trajectory planning. Because of these challenges and the large number of papers in the field of exoskeleton control, this paper addresses the trajectory-planning problem. The purpose of this literature review is to explore trajectory-planning methods for exoskeleton control, especially for the upper limbs. We also identified challenges related to trajectory planning for upper-limb exoskeletons.

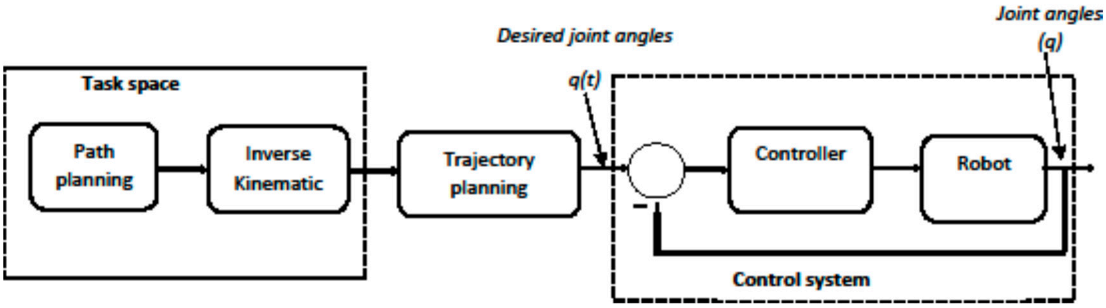

Figure 1. Trajectory planning based on Cartesian motion planners.

2. Materials and Methods

A search ranging from 1999 to 2020 was conducted in Ei Compendex, Inspect-IET, Web of Science, PubMed, ProQuest, and Science-Direct using the following search strategy. "Trajectory planning", "upper limb", and "exoskeleton" were used and combined as high-level keywords. Additionally, "trajectory planning" or "motion planning" was combined with the following keywords: "rehabilitation", "humanlike motion", "upper extremity", "inverse kinematic", and "learning machine". These keywords were selected after performing an initial search to ensure that there were no areas within this field that were excluded due to inadequate keyword selection. Abstracts and full texts were then collaboratively evaluated by the authors for direct relevance to the topic and scope of this study. A total of 67 papers were discovered.

Figure 2 shows that analytic expression-based (polynomial methods, sigmoid functions, Fourier series) and computational or optimization-based methods are the most prevalent methods used for the trajectory planning of upper-limb exoskeletons. The number of studies involving trajectory planning for exoskeleton robots has increased, especially over the last three years (Figure 3). Of the papers collected involving trajectory planning, 32.07% were from 2018 to 2020 and were related to analytic expression-based and optimization-based methods. This evolution justifies the growing interest in this topic.

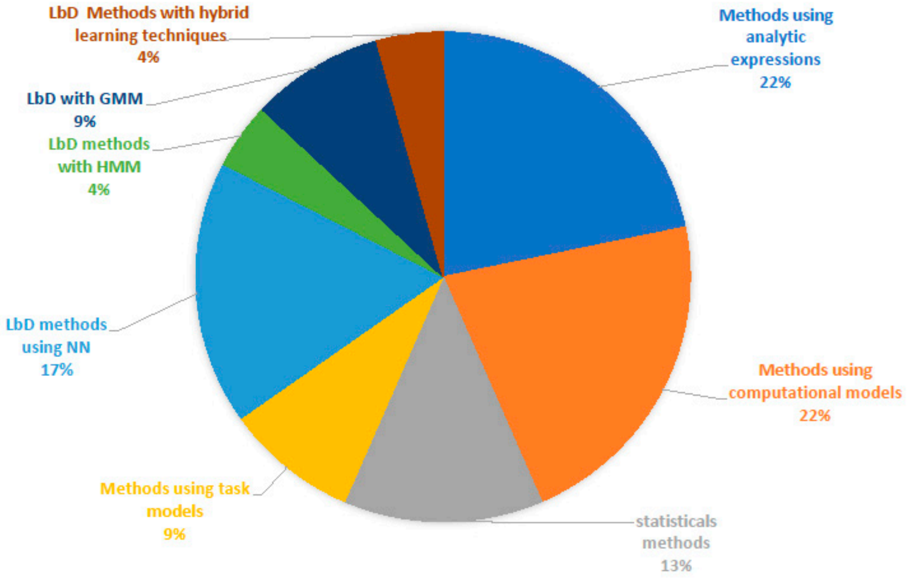

Figure 2. Trajectory-planning methods in the literature, 1999–2020.

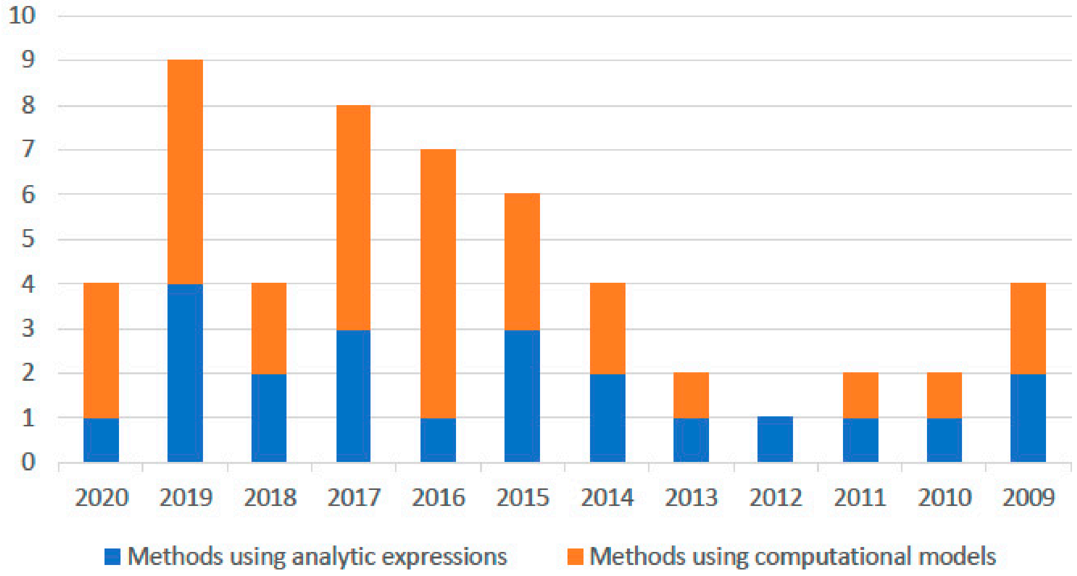

Figure 3. Papers retrieved from January 2009 to August 2020 for the two most common trajectory-planning methods.

3. Review of Trajectory Planning

First, the trajectory-planning problem consists of finding a relationship between two elements belonging to the domains of space and time. Thus, the trajectory is usually expressed as a parametric function of time, providing the corresponding desired position at each moment. For exoskeletons, trajectory planning can either be carried out in task space or in joint space, depending on the control method. It can also be a point-to-point or predefined path. Inverse and forward kinematics are very important for moving from one space to another. Inverse kinematics consists of finding the Cartesian position of the end effector given the angular position of a joint. As illustrated in Figure 1, trajectory planning provides the desired trajectories of the end effector in task space or the desired joint angles in joint space. We highlight the high stakes of inverse kinematics in the trajectory-planning problem. When planning in task space, the Cartesian trajectory is first transformed into a sequence of joint displacements through inverse kinematics (IK). Then, to complete the planning process, the motion profiles for each joint are generated by interpolation while considering a specific set of constraints (in accordance with the design requirements) [20] or by approximation [21]. The common method of interpolation is by using third- or fifth-degree polynomials [9,12,22,23]. In concrete terms, the desired motion is defined by only assuming initial and final points or by also considering a set of intermediate via-points that must be properly interpolated. These polynomials make it possible to describe the trajectory between any two positions. For example, Li et al. [12], and Ghobadhi et al. [24] employed fifth-order polynomials for the trajectory planning of an upper-limb rehabilitation exoskeleton. There are many potential advantages to using B-spline as the interpolation function. Similarly, when trajectories are planned in joint space, the joint positions obtained manually or by inverse kinematics are interpolated. Planning in joint space involves a higher computational cost because the transformation from Cartesian to joint-space coordinates takes place in real-time since control happens at the joint level. However, planning in joint space is faster. The difficulty of trajectory planning in task space is related to the resolution of the inverse kinematics problem. Depending on the control method and the space in which constraints are given, planning in each space is applicable. Several methods are available for planning the desired movement of upper-limb exoskeletons, and they can be classified into two main approaches.

3.1. Approaches Based on Cartesian Motion Planning

This approach is based on path planning (in task space) and the resolution of inverse kinematics (IK). Inverse kinematics is necessary to compute joint motion for exoskeletons. Cartesian coordinates (all points or knot points) are transformed into joint angles by using inverse kinematics. Inverse kinematics can be resolved using Jacobian-based methods [12], which exploit the available degrees of freedom (DOFs) of the robot's kinematic chain to achieve the desired end-effector pose. However, since the kinematic redundancy of the human arm allows for multiple arm configurations to complete the task in the three-dimensional workspace, it is difficult to obtain a nonunique solution for inverse kinematics in the human–robot workspace [25,26]. To overcome this difficulty, an alternative approach based on swivel-angle computation is often used [26,27]. In [25], the resolution of the redundancy through computation of the swivel angle made it possible to obtain unique solutions for the joint space. The application of trajectory planning based on path planning and inverse kinematics was presented in many studies [12,28,29]. In [28], the authors proposed a general kinematic model with Denavit–Hartenberg (DH) convention for the path generation of upper-limb exoskeleton robots. Li et al. used the reverse coordinate method to complete the inverse kinematics solutions; they also proposed a new multicubic polynomial-interpolation method for planning joint trajectories [12].

The Cartesian trajectory is commonly obtained by kinematic analysis [30], the Jacobian equation, geodesic curves [27,31], DH convention, and other methods. Another way to define the Cartesian trajectory is by using motion models on the basis of the kind of task, such as circular motions [32] and periodic movements [33,34]. These motion models can be derived from the results of the trajectory transformation corresponding to each joint movement during an ADL task. Accordingly, Meng et al. proposed circular-arc trajectory planning for an upper-limb-exoskeleton rehabilitation robot [13]. For exoskeletons, once Cartesian trajectories and joint positions (through the inverse kinematics) are obtained, trajectory planning is completed in joint space using a polynomial-interpolation method. Although polynomial methods, of which so-called Bezier polynomials is the most popular, are a regular method for planning the desired trajectory of robotic systems, they require significant computing resources. For this reason, an alternative strategy based on the use of sigmoidal functions is proposed to construct reference trajectories [21]. However, such trajectories, which use mathematical expressions, are not exactly natural human trajectories since they are not related to the principles governing the control of human arm movements (such as optimization theory [35] and isochrony principle [32]). Thus, the trajectory can be planned in joint space by formulating a mathematical model of human motor behavior in order to plan the desired trajectories in a way similar to that of humans. Many mathematical models of human behavior, known as optimization-based methods, were proposed for humanlike motion planning [27,31,36–38]. These methods assume that human motions are generated by optimizing a known cost function. The most common of these models are the minimum-jerk [25,32,39], minimum-torque [40,41], inertia-like [31,42] and minimum-potential-energy [37] models. In [25], the spatial trajectory of the end effector was generated on the basis of a minimum-jerk model. A minimum-jerk approach was also adopted in [24,39,43,44]. In [37], Li et al. exploited the minimization of potential energy. In the literature, the choice of the cost function to be optimized in typical approaches to generating human likeness in robotic motions is a hotly debated topic. In order to not have to choose a cost function, Averta et al. exploited functional principal-component analysis (fPCA) for humanlike motion planning [42]. Their method was to extract principal motion patterns from recorded data by using fPCA and optimize the weights of a reduced set of these components. fPCA is the functional equivalent of principal-component analysis (PCA) in the temporal domain, and Tang et al. confirmed that kinematic synergies (principal components) could be used for exoskeleton motion planning [45]. They also proved that different principal components contributed to the motion trajectory. Another computational approach based on motion primitives (principal components) was used in [46–50]. However, these typical computational methods used to generate humanlike motions in the literature are only appropriate for simple point-to-point movements with a limited range of motion because they assume a fixed shoulder. Consequently, they are not suitable for most ADLs, during which the center of the shoulder joint moves significantly. Faced with this drawback of the fixed shoulder assumption, Soltani-Zarrin et al. proposed a generation method of exoskeleton path that considers the effect of scapulohumeral rhythms. In this work, the authors adopted geodesic curves to generate the joint angular path [27]. Rather than planning the trajectory offline, which optimizes the trajectory to perform a specific task in a structured environment, Frisoli et al. proposed a method of bounded-jerk online trajectory planning [51]. In their planning method, the generation of upper-limb-exoskeleton trajectory paths relies on determining the immediate motion intention of the user through gaze exploitation. Moreover, their method not only permits online humanlike trajectories but also guarantees the synchronization of joints during multi-DOF movements. Another advantage of such online trajectory planning is its ability to adapt to changes in the environment [7].

3.2. Approaches Based on Learning by Demonstration (LbD)

Learning by demonstration (LbD), also known as programming by demonstration (PbD), is sometimes employed in robotic programming for complex and nonstrict motion trajectories [52]. LbD consists of two stages (Figure 4): (i) a learning phase that first acquires behavior data and encodes them through a learning model, and (ii) a reproduction phase that uses appropriate control models to reproduce similar behavior [15,53,54]. Planning based on LbD is an alternative approach to the traditional approach to path planning and IK illustrated in the previous section since it either makes it possible to avoid motion planning in Cartesian space (as in [11]), or it helps to find a unique inverse kinematics solution (as in [18]). However, this approach requires learning the target joint configuration. For upper-limb exoskeletons, the generation of reference trajectories is grounded in learning models, the most common of which are neural networks (NNs) [8,11,54], hidden Markov models (HMM) [55], dynamic-motion-primitive (DMP) models [45], and Gaussian mixture models [14,56]. Concretely, the idea behind LbD is to extract an adequate control law from demonstrations of human motion during ADL tasks.

In the literature, many planning methods based on LbD were proposed for redundant and nonredundant exoskeleton robots [11,14,45,54–57]. These methods usually utilize machine-learning methods (supervised and reinforcement learning [58]) to learn the IK of exoskeletons robots. For tasks with little to no interaction with the environment, such as writing, the observation of human motion (demonstration) makes it possible to train a movement model that can replicate the task. LbD is also used for complex tasks involving interactions with the environment, where the robot learns using HMMs. In [11], a motion-planning method based on LbD was adopted to generate reference trajectories in joint space. The authors used an NN to learn motion features (DMP) and the robot's inverse kinematics. Indeed, depending on the task and object position, an NN trained through the Levenberg–Marquart supervised learning algorithm can provide distinctive features (DMP parameters) and target joint positions. These DMP parameters are then processed to provide the exoskeleton with reference joint trajectories. In [54], three trained artificial NNs (ANNs) were combined in the form of a closed-loop model that generated elbow angles similar to experimentally recorded trajectories. Garrido et al. adopted a planning approach based on LbD [55]. From human demonstrations, they generated the desired joint trajectory by using a learning method based on an HMM [55]. A supervised artificial neural network was employed in [18] to conduct the motion-trajectory models from a set of training patterns collected through a motion capture system. Other studies employing the proposed planning method based on LbD were [45,56], wherein a DMP and a Gaussian mixture model, respectively, were used to generate trajectory planning for reaching movements.

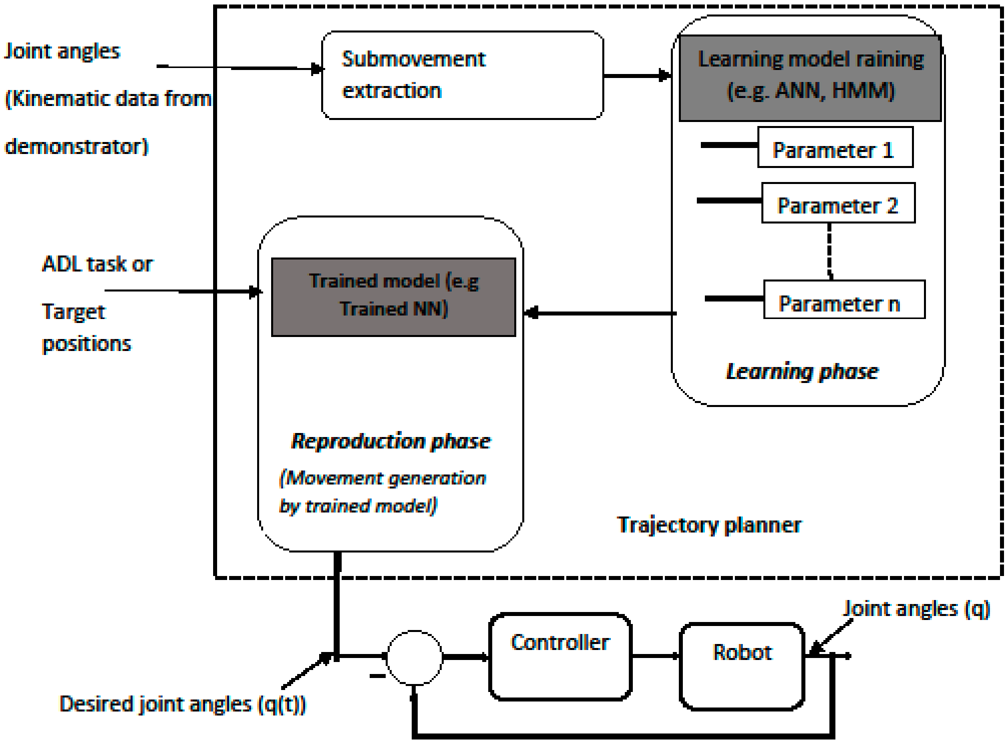

Figure 4. Adapted from [14]. Trajectory planning based on learning by demonstration. ADL: activities-of-daily-life; NN: neural network; ANN: artificial NN; HMM: hidden Markov model.

4. Discussion

4.1. Challenges in Trajectory Planning of Exoskeleton Robots

Great advances in robotics have enabled the use of exoskeletons for functional movements with repeatability and intensity in rehabilitation and for assistive goals. In addition, with trajectory planning, their usage can be extended to home usage. Despite the advantages of these robotic devices, there are still challenges associated with their use. In this section, we discuss the challenges that make trajectory planning more difficult.

Due to the physical coupling between the human body and the exoskeleton, typical strategies for determining the desired trajectory to be tracked by the exoskeleton during complex tasks consist of reproducing the movement of the human limb. However, it is difficult to accurately replicate human kinematics with robots because of the morphologic variability between individuals and the complexity of joint kinematics [59]. The differences between kinematic chains, referred to as "kinematic incompatibility", of human limbs and robots lead to hyperstaticity or overconstraint [60]. As a result, this can hinder the security and comfort of the user. In the literature, it was indicated that adding more DoFs into the configuration of an exoskeleton improves kinematic compatibility [61]. For this goal, there are two configurations—self-tracing and alignment-free exoskeletons of the joint axis (center)—that are exploited for upper-limb exoskeletons [62]. In the self-tracing configuration, an appropriate number of DoFs of active or passive joints are added in series into the main exoskeleton configuration. The alignment-free configuration consists of introducing an appropriate number of passive joints into the link between the exoskeleton and the user's upper limb. Both exoskeleton configurations present their own advantages and problems that are not addressed in this paper. With the aim of combining the benefits of both configurations, Li, Jianfeng et al. recently proposed a 4-DOF self-aligning mechanism for upper-limb exoskeletons [60]. Moreover, the kinematic redundancy of the human arm poses challenges with respect to joint-space trajectory planning for upper-limb exoskeletons. The essence of the issue here is the inverse kinematics problem, whose solution is complicated when the number of DoFs at the joint level exceeds one at the hand level. A review indicates that two approaches are typically exploited to resolve kinematic redundancy in exoskeletons [63]. In the first approach, angular excursions at the joint are minimized in some way. On the basis of this concept, some previous studies used methods based on the calculation of swivel angles [25,47]. The second approach relies on the relationship between the degrees of freedom. In addition to these limits, two main challenges were identified in trajectory planning for exoskeleton robots [7]. The first is trajectory optimization [7,10,64], also known as the optimal-control problem [64,65]; it is challenging to optimize a trajectory in order to suit a user's motion profile [66]. This difficulty results from user motion profiles being unique to the kinematics and dynamics of the wearer and to environmental settings. There are two approaches to solving the trajectory-optimization problem for exoskeletons, offline, and online trajectory optimization. In the online approach, there are two possibilities. The first is that optimization is run in the background, and the trajectory is periodically updated, leading to the low computational performance and poor reaction time of the trajectory planner. An alternative possibility consists of using a simplified model to accelerate calculation, but there is no guarantee of convergence since the definition of a trajectory-optimization problem entails the definition of the dynamic environment [64]. The second approach, offline trajectory optimization, makes it possible to overcome the computational cost and guarantees convergence. In the literature, there are (i) methods that use a function approximation through trajectory learning from a set of optimal trajectories, such as an optimal-control method that was used in [10]; and (ii) other methods of optimal control based on nonlinear programming, as proposed in [46,65] for optimal-control problems.

Lastly, the final challenge identified in [7] is associated with trajectory planning in an unstructured environment. Trajectory planning in an environment that is subject to changes is difficult because exoskeletons do not have complete information regarding their surroundings [7]. In such situations, the trajectory-planning method must be able to manage variability in the environment while preserving performance. In the literature, most trajectory-planning methods proposed for upper-limb exoskeletons rely on offline planning (Tables 1 and 2); this is the state of the art in trajectory planning. Moreover, in online trajectory planning for exoskeleton robots, the generation of trajectories relies on the immediate intentions of the user, and its extraction is carried out by sensors, the limits of which are well-known.

4.2. Methods Applied to the Currently Available Exoskeleton Robots

In robotic rehabilitation, exoskeletons have been developed to assist patients after stroke by moving their impaired limb through a predefined trajectory. Trajectory generation is a fundamental topic in the design of these robots because of the growing number of stroke patients. The usual approach with most exoskeletons is to use a prerecorded trajectory as a look-up table. There are some limitations to this method, including data storage limitations and poor tuning relating to the motion's parameters. According to the classification in Section 3, two motion planning approaches are used for current upper limb exoskeletons: the traditional approaches based on Cartesian motion planning and inverse kinematics and approaches based on learning by demonstration (LbD).

In the first approach (Table 1), cubic and polynomial methods are the most used in current exoskeletons. The limitation of these methods is that they require the calculation of too many parameters (polynomial coefficients), and the generated trajectories are un-natural. An alternative is the optimization-based methods: (i) based on optimization of a cost function such as minimum-jerk, minimum-torque-change, and inertia-like models; (ii) based on the extraction of principal motion patterns like principal component analysis (PCA) (as in [45]) and functional PCA (fPCA) (as in [42]), which were used to derive kinematic synergies. Optimization-based methods usually involve hypotheses on the motion generation that limit the variability of the planned movement. These methods generate humanlike motions, but they are only appropriate for simple point-to-point movements with a limited range of motion. So, they are unsuitable when applied to complex tasks like those of activities-of-daily-life (ADLs).

To produce the greatest effect in robotic rehabilitation, exoskeleton robots require the online trajectory capacity to operate in a real environment. Trajectory planning methods in the first trajectory are unsuitable for unstructured environments like in the real world. This is why research groups have been interested in the second type of approach (Table 2), imitation-learning methods (LbD methods). These methods rely on trajectory learning, where robots learn human skills from demonstrations and later reproduce the movement through a trained model or a function approximation. Deconvolutional neural networks (DNN) (as in [10]), multilevel convolutional neural networks (as in [8]), and artificial neural networks (ANNs) (as in [18,54]) are used for function approximation; Gaussian mixture models (GMM) (as in [14,56]), hidden Markov models (HMM), and dynamic-motion-primitive (DMP) models (as in [11]) are used as models for upper-limb exoskeleton robots in the training phase. The training is achieved using expectation-maximization (EM) algorithms (as in [14,56]) and the Levenberg-Marquart algorithm (as in [11,54]). The LbD methods have the advantage of avoiding the calculation of the inverse kinematics. These methods have high computational costs, but this limit no longer exists once the learning phase is complete. However, demonstrations in joint space are strongly time-dependent. This issue is considered in the literature using methods based on dynamic time warping (DTW), which destroys the accuracy of the generated trajectories. Garrido et al. used Lloyd's algorithm to encode input signals and a modified HMM to more accurately plan trajectories in joint space. Although the study has not been tested on patients, we think that this method of generating the desired trajectory in joint space should be explored. Our opinion is motivated by two reasons: firstly, because there are few works in joint space; secondly, because trajectory learning is still unexplored for exoskeleton robots in comparison to policy learning.

The studies presented in this review are summarized in Tables 1 and 2. The last column of these tables indicates if the proposed methods have been tested on humans (patients or healthy subjects) and, if so, the number of patients. From the results presented in these columns, we observed that 45.4% of these studies conducted testing only by numerical simulations, 31.8% performed testing on healthy subjects, and only 9% on patients (Table 3). The tests have been performed on patients in two studies [11,43]. Considering the presented studies, only a few robotic devices for upper limb rehabilitation have been tested in the clinical environment, likely due to the lack of standardized data measurement and evaluation procedure, since verifying their efficacy is a critical issue related to robotic therapy. In addition, these tests were performed on a small number of subjects (maximum of four subjects). Consequently, we report only two kinds of problems encountered during the clinical tests: (i) control problems: test provided low accuracy, which could limit the application of the proposed method to a given domain [11,47]; (ii) generalization problems: low generalization capacity and low accuracy [43].

From a rehabilitation point of view, the efficacy and reliability are still clinically unproven, so we hope that researchers will focus on advancing rehabilitation robotics from technical laboratories to clinics. This would help increase the use of robotic devices in rehabilitation therapy. Moreover, careful attention needs to be paid to clinical tests and the effectiveness of diagnosed stroke patients because clinicians will be interested in devices with proven efficacy.

Table 1. Summary of studies on trajectory planning for upper-limb exoskeletons (approaches based on Cartesian motion planning). DoF, degrees of freedom.

Trajectory Scheme	Planning Methods	Trajectory-Generation Method	Planning Type	Dimensionality (DoFs)/Exoskeleton Design	Advantages and Limitations	Applications	Tested on Patients/Healthy Subjects (Yes/No)
(Li, 2019) [12] Cartesian coordinates.	Multicubic polynomial-interpolation Method.	Finds angular positions through inverse kinematics (IK; using reverse coordinate method).	Offline.	Six DoFs.	Pros: quickly giving a solution and the flexibility of being able to refine human trajectories.	Reach-to-grasp movements.	No
(Wang, 2019) [25] Recorded data (to calculate swivel angles).	Planning based on minimum-jerk model.	Inverse kinematics with swivel-angle + minimum-jerk method.	Offline.	Five DoFs.	Provides optimal reference trajectory (because generated joint trajectories are humanlike motions).	Reaching and reach-to-grasp movements.	No
(Ballesteros 2019) [21] Recorded data.	Planning based on sigmoidal functions.	Predefined trajectory by therapist (desired articular positions).	Offline	-	Simple and improves execution time in comparison to planning based on polynomial methods.	Assistance.	Yes (two healthy subjects)
(Frisoli, 2013) [51] Recorded data + machine system vision (to automatically identify and localize target online).	Bounded-jerk planning + motion synchronization.	Bounded-jerk method.	Online.	Five DOFs.	Allows for full synchronization among joints during multi-DOF movements. Generates humanlike motions.	Reaching tasks.	Yes (four healthy subjects)
(Meng, 2018) [13] Recorded (circular-motion) data.	Task-based planning method (circular-arc planning).	Task analysis in order to find a planning task (circular motion).	Offline.	Four DoFs.	Well-suited to rehabilitation because such motion predefinition can clarify the training target.	Rehabilitation tasks.	Yes (-)

Table 1. *Cont.*

Trajectory Scheme	Planning Methods	Trajectory-Generation Method	Planning Type	Dimensionality (DoFs)/Exoskeleton Design	Advantages and Limitations	Applications	Tested on Patients/Healthy Subjects (Yes/No)
(Soltani-Zarring, 2017) [27] Captured kinematic data + therapeutic values.	Optimization method based on Riemannian geometry.	Trajectory generated by geodesic curves.	Offline.	Four DoFs.	Suited to most upper-limb motions such as complex activities of daily life (ADL) tasks during which the center of the shoulder moves significantly.	Rehabilitation and ADL tasks.	No
(Kim, 2014) [75] recorded human data.	Fourier approximation.	Joint trajectories obtained through motion builder (AutoCAD).	Offline.	Four DoFs.	Pro: low cost.	Joint and muscle rehabilitation: frozen shoulder.	Yes (One healthy subject) two types of problem: (i): Transmission system (vibrations due to reduction gears and compliance was noted) (ii) Customization problem: height adjustable system
(Xie, 2019) [28]	Cubic and quintic polynomial methods.	Kinematic resolution by modeling human upper limbs: robot parameters substituted into model kinematics equation.	Offline.	Four DoFs.	Proposed method reduces the process of kinematics calculation.	Upper-limb rehabilitation (taking task); ADL tasks.	Yes (one healthy subject)
(Meng 2014) [34]	Planning based on motion models (cycloid movement).	Cycloid law.	Offline.	Three DoFs.	con: method tested in too basic a system.	Hand rehabilitation: index-finger movements.	No

Table 1. *Cont.*

Trajectory Scheme	Planning Methods	Trajectory-Generation Method	Planning Type	Dimensionality (DoFs)/Exoskeleton Design	Advantages and Limitations	Applications	Tested on Patients/Healthy Subjects (Yes/No)
(Li-Zhan, 2020) [37]	Planning based on potential-energy minimization.	Inverse kinematics (IK) resolution based on zeroing dynamics method (ZD): IK + ZD.	-	Six DoFs	Proposed method guarantees elegant convergence of inverse kinematics solution.	Industrial application: manipulation tasks.	No
(Enya, 2011) [43]	Planning based on minimum-jerk.	Minimum-jerk trajectory.	Offline.	Four DoFs.	Method efficaciously plans reaching angle when tested by a patient with brachial plexus injury.	Assistance: reaching movements.	Yes (one patient)

Table 2. Summary of studies on trajectory planning for upper-limb exoskeletons (approaches based on learning by demonstration (LbD)).

Trajectory Scheme	Motion Features/ Parameters	Learning Models	Training Methods	Trajectory Generation (Reproducing Phase)	Advantages and Limitations	Applications	Tested on Patients/Healthy Subjects (Yes/No)
(Naghibi, 2020) [54] Angular trajectories + extracted minimum-jerk submovements	Amplitude, duration, and start time of minimal submovements.	Three distinct artificial neural networks (ANNs).	Levenberg–Marquart algorithm.	Predictive model consisted of three ANNs (generates angular joint profiles).	Cons: method requires phasing of ADL task (360 phases here) Pros: predictive model can correct prediction error, so generated movements are accurate.	ADL movements such as eating and drinking.	No

Table 2. *Cont.*

Trajectory Scheme	Motion Features/ Parameters	Learning Models	Training Methods	Trajectory Generation (Reproducing Phase)	Advantages and Limitations	Applications	Tested on Patients/Healthy Subjects (Yes/No)
(Lauretti, 2018) [11] Recorded trajectories	Dynamic-motion-primitives (DMP) parameters.	Neural-network (NN) training + DMP computation.	NN trained by Levenberg–Marquart supervised-learning algorithm.	DMP with well-defined landscape attractor, allowing for replication of movement by weighted sum of optimally spaced Gaussian kernels.	It can work in unstructured environments. The accomplishment of tasks in a feasible workspace is guaranteed.	Reaching movements, grasping.	Yes (4 patients) One problem: Test provided higher position errors, so the proposed method is not applicable to domains that do not require a very high accuracy
(Garrido, 2016) [55] Trajectory database.	Codebook (discrete values) + key points	Modified hidden Markov model (HMM) (discrete HMM).	Lloyd's algorithm.	Trained modified HMM.	It improves accuracy compared with dynamic time-warping methods.	Writing task.	No
(Sabbaghi, 2014) [14] Captured monocular images of 3D human poses.	Gaussian components.	Gaussian mixture model (GMM).	Expectation-maximization algorithm (EM)	Gaussian mixture regression (GMR)	Enables gesture learning using only a single camera without equipping demonstrator with color or patches.	Healthcare and education.	-
(Deng, 2020) [56] Sensed interaction force.	GMM parameters.	GMM.	GMM trained using expectation-maximization (EM) algorithm offline.	Interaction-force generation by using Gaussian mixture regression (GMR).	Considers physical constraints of the exoskeleton and variability among multiple demonstrations.	Manipulation tasks: drawing.	Yes (two healthy subjects)

Table 2. *Cont.*

Trajectory Scheme	Motion Features/ Parameters	Learning Models	Training Methods	Trajectory Generation (Reproducing Phase)	Advantages and Limitations	Applications	Tested on Patients/Healthy Subjects (Yes/No)
(Tao, 2020) [8] Recorded video (depth information and color information) by Kinect camera.	Human pose.	Multilevel convolutional NN.	Human-pose recognition algorithm OpenPose (open-source library).	Trajectory extraction (by OpenPose) + threshold-filtering method.	Simpler and more convenient for rehabilitation robots because video-based teaching greatly reduces difficulty in redesigning rehabilitation trajectories.	Upper-limb rehabilitation.	Yes (one healthy subject)
(Chung, 2012) [18] Recorded data	(i) Distance between shoulder and end effector, and elbow height; (ii) angle between shoulder and end effector, angle between shoulder and end effector with respect to elbow, and distance between end effector and shoulder.	ANN model.	Supervised ANN approach.	Trained ANN model used to resolve inverse kinematics problem.	Performs similar motions to human motions.	Writing, hand waving, beating, and throwing a ball.	No
(Duburcq, 2020) [10]	Optimal trajectories generated through trajectory optimization.	Deconvolutional neural network (DNN) as function approximation.	Guided-trajectory-learning (GTL) algorithm.	Online trajectories generated by trained function approximation DNN.	Produces more accurate and reliable online trajectories based on LbD by guaranteeing prediction feasibility.		No

Table 2. *Cont.*

Trajectory Scheme	Motion Features/ Parameters	Learning Models	Training Methods	Trajectory Generation (Reproducing Phase)	Advantages and Limitations	Applications	Tested on Patients/Healthy Subjects (Yes/No)
(Averta, 2020) [42] Recorded joint trajectories.	Functional principal components (fPCs).	Functional principal component analysis (fPCA) characterization.	fPCA.	Optimization of fPCS weights.	Simplicity: only three fCPs represented 80% of the motion trajectory.	Rehabilitation, assistance.	No
(Tang, 2019) [45] Recorded kinematic data + (angular velocity matrix).	Spatiotemporal kinematic synergies (principal components).	Kinematic-synergy analysis	PCA algorithm.	Singular-value-decomposition (SVD) algorithm.	Pros: simplifies motion planning because only the first four principal components were enough to represent the motion.	Rehabilitation: multi-DoF movement in upper limbs.	No
(Liu, 2018) [47].	Postural synergy.	Kinematic-synergy analysis.	PCA method.	Dramatically simplified mechanism system.	Simple planning method.	Rehabilitation, assistance: reaching movements.	Yes (15 healthy subjects) Problems: low generalization capacity, low accuracy (i) The subjects chosen for the tests were among subjects who participated in the data collection experiments; (ii) The exoskeleton robot did not replicate the supination-pronation motions of the elbow joint with high accuracy

Table 3. Kind of tests performed by presented studies.

Studies Tested on Healthy Subjects	Studies Tested on Patients	Untested on Human Beings	Others	Effective
7 31.8%	2 9%	10 45.4%	3 13.6%	22

4.3. Clinical Needs in Robotic Rehabilitation of Upper Limbs

Besides the efficacy/effectiveness aspect [67] mentioned in Section 4.2, other needs of the clinical practice should be considered by the researchers when designing rehabilitation robots. These clinical needs can be grouped into four aspects [68]:

(1) Psychological: The patient should not be afraid of robotic rehabilitation devices. They should not replace the therapist but make their job easier.

(2) Medical: The robot should be adaptable to the human limb in terms of segment lengths, range of motion, and degrees of freedom (DoFs) [68,69]. To adapt the robot to different human bodies (sizes and weights), the segments of exoskeletons should be adjustable. A robot with a high degree of freedom (DoF) offers a wide variety of movements with many anatomical joint axes [68]. Modularity would be welcome in the medical aspects because the modular prototype, as the one previously proposed [12], can offer single-joint rehabilitation training, as well as multi-joint composite training. This feature increases the flexibility of exoskeleton robot applications in rehabilitation [70].

(3) Ergonomic: The design of an upper limb exoskeleton robot must allow some additional space for the patient.

(4) Control: The robot should be back-drivable. The weight of the robot should not be felt by the patient, and they should be able to move the robotic device. Assist-as-needed control is the most common approach used in robotic rehabilitation devices [25,56,71]. This method allows the patients to initiate the movement on their own and to be partially assisted when they start regaining their lost motor function (active therapy) [70,72]. The need here is control sharing, which is the ability to control the forces applied by the robot [68]. A previously proposed prototype [43] is back-drivable (instrumented compliance). There are two kinds of back-drivability [73], the most advantageous of which, mechanical back-drivability, is strongly dependent on the transmission mechanism of the exoskeleton [73]. Consequently, back-drivability is an important clinical need that must be considered by researchers.

The spasticity measurement is a potential clinical need because spasticity is the most common symptoms of stroke patients [74].

5. Conclusions

Trajectory planning is an important part of the control of intelligent robotic systems such as exoskeleton robots. It has been proven in the literature that trajectory planning helps to better control the movement of exoskeleton joints and improve the ADLs of stroke patients, helping them return to society and their occupation. This review explores the methods of trajectory planning for exoskeleton control, especially at the upper limbs. From 67 relevant papers found in the literature, we categorized trajectory-planning methods into two main approaches:

1. Approaches based on Cartesian motion planning and inverse kinematics using polynomial-interpolation or computational methods such as minimum-jerk, minimum-torque-change, and inertia-like models.
2. Approaches based on learning by demonstration (LbD) using learning models such as neural networks (NNs), HMMs, Gaussian mixture models, and DMP models.

As shown in this review, each method has its limitations. In the first approach, generated trajectories by methods based on polynomial trajectories are not natural. Moreover, optimization-based methods that rely on humanlike motions are only appropriate for simple point-to-point movements with a limited range of motion. They are not suitable when applied to complex tasks like those of ADLs. Second, imitation-learning methods have high computational costs.

In this paper, we also discussed challenges associated with trajectory planning: kinematic redundancy and incompatibility, and the trajectory-optimization problem. For upper-limb exoskeletons, methods based on the computation of swivel angles and other approaches commonly rely on the relationship (e.g., coordinated or synergistic) between degrees of freedom to resolve kinematic redundancy for exoskeletons. Moreover, two general solutions, namely, the self-tracing and alignment-free configurations of the joint axis, which add the appropriate number of extra degrees of freedom to the mechanism, are employed to improve kinematic incompatibility between humans and exoskeletons.

Although online trajectory planning can be adapted to manage the variability of the environment, very few of the proposed methods for planning trajectories in upper-limb exoskeletons generate offline trajectories. Future work will focus on online trajectory planning and optimal control because very few online methods were found during this study. In future works, tests should be completed on more patients. In any study on exoskeleton robots, we recommend considering the clinical needs presented in this review.

Author Contributions: Conceptualization, C.N. and S.A.; research and original draft preparation, C.N.; review, editing and revised draft preparation, C.N. and M.R.; supervision M.R. and S.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

ADL	activities-of-daily-life
ANN	artificial neural network
DH	Denavit-Hartenberg
DMP	dynamic motion primitives
DNN	deconvolutional neural network
DoF(s)	degree(s) of freedom
DTW	dynamic time warping
(fPCs)	functional principal-components
fPCA	functional principal-components analysis
GMM	Gaussian mixture model
GMR	Gaussian mixture regression
HMM	hidden Markov model
IK	inverse kinematics
LbD	learning by demonstration
M-DH	modified Denavit-Hartenberg
NN	neural network
PCA	principal-components analysis
PbD	programming by demonstration

References

1. Boubaker, O. Chapter 7—Medical robotics. In *Control Theory in Biomedical Engineering*; Boubaker, O., Ed.; Academic Press: Cambridge, MA, USA, 2020; pp. 153–204. [CrossRef]
2. Bertani, R.; Melegari, C.; De Cola, M.C.; Bramanti, A.; Bramanti, P.; Calabrò, R.S. Effects of robot-assisted upper limb rehabilitation in stroke patients: A systematic review with meta-analysis. *Neurol. Sci. Off. J. Ital. Neurol. Soc. Ital. Soc. Clin. Neurophysiol.* **2017**, *38*, 1561–1569. [CrossRef] [PubMed]

3. Zengin-Metli, D.; Özbudak-Demir, S.; Eraktaş, İ.; Binay-Safer, V.; Ekiz, T. Effects of robot assistive upper extremity rehabilitation on motor and cognitive recovery, the quality of life, and activities of daily living in stroke patients. *J. Back Musculoskelet. Rehabil.* **2018**, *31*, 1059–1064. [CrossRef] [PubMed]
4. Mehrholz, J.; Pohl, M.; Kugler, J.; Elsner, B. The Improvement of Walking Ability Following Stroke. *Dtsch. Arztebl. Int.* **2018**, *115*, 639–645. [CrossRef] [PubMed]
5. Daunoraviciene, K.; Adomaviciene, A.; Grigonyte, A.; Griškevičius, J.; Juocevicius, A. Effects of robot-assisted training on upper limb functional recovery during the rehabilitation of poststroke patients. *Technol. Health Care Off. J. Eur. Soc. Eng. Med.* **2018**, *26*, 533–542. [CrossRef]
6. Gupta, A.; Singh, A.; Verma, V.; Mondal, A.K.; Gupta, M.K. Developments and clinical evaluations of robotic exoskeleton technology for human upper-limb rehabilitation. *Adv. Robot.* **2020**, *34*, 1–18. [CrossRef]
7. Miskon, M.F.B.; Yusof, M.B.A.J. Review of trajectory generation of exoskeleton robots. In Proceedings of the 2014 IEEE International Symposium on Robotics and Manufacturing Automation (ROMA), Kuala Lumpur, Malaysia, 15–16 December 2014; pp. 12–17.
8. Tao, T.; Yang, X.; Xu, J.; Wang, W.; Zhang, S.; Li, M.; Xu, G. Trajectory Planning of Upper Limb Rehabilitation Robot Based on Human Pose Estimation. In Proceedings of the 2020 17th International Conference on Ubiquitous Robots (UR), Kyoto, Japan, 22–26 June 2020; pp. 333–338.
9. Ali, S.A.; Annuar, K.A.M.; Miskon, M.F. Trajectory planning for exoskeleton robot by using cubic and quintic polynomial equation. *Int. J. Appl. Eng. Res.* **2016**, *11*, 7943–7946.
10. Duburcq, A.; Chevaleyre, Y.; Bredech, N.; Boéris, G. Online trajectory planning through combined trajectory optimization and function approximation: Application to the exoskeleton Atalante. *arXiv* **2019**, arXiv:1910.00514.
11. Lauretti, C.; Cordella, F.; Ciancio, A.L.; Trigili, E.; Catalan, J.M.; Badesa, F.J.; Crea, S.; Pagliara, S.M.; Sterzi, S.; Vitiello, N. Learning by demonstration for motion planning of upper-limb exoskeletons. *Front. Neurorobotics* **2018**, *12*, 5. [CrossRef]
12. Li, G.; Fang, Q.; Xu, T.; Zhao, J.; Cai, H.; Zhu, Y. Inverse kinematic analysis and trajectory planning of a modular upper limb rehabilitation exoskeleton. *Technol. Health Care* **2019**, *27*, 123–132. [CrossRef]
13. Meng, Q.; Shao, H.; Wang, L.; Yu, H. Task-Based Trajectory Planning for an Exoskeleton Upper Limb Rehabilitation Robot. In Proceedings of the International Conference on Man-Machine-Environment System Engineering, Nanjing, China, 20–22 October 2018; pp. 141–149.
14. Sabbaghi, E.; Bahrami, M.; Ghidary, S.S. Learning of Gestures by Imitation using a Monocular Vision System on a Humanoid Robot. In Proceedings of the 2014 Second Rsi/Ism International Conference on Robotics and Mechatronics, Tehran, Iran, 15–17 October 2014; IEEE: New York, NY, USA, 2014; pp. 588–594.
15. Liu, N.; Liu, Z.; Cui, L. A Modified Cartesian Space DMPs Model for Robot Motion Generation. In Proceedings of the International Conference on Intelligent Robotics and Applications, Shenyang, China, 8–11 August 2019; pp. 76–85.
16. Yu, W. Chapter 1—Preliminaries. In *PID Control with Intelligent Compensation for Exoskeleton Robots*; Yu, W., Ed.; Academic Press: Cambridge, MA, USA, 2018; pp. 1–12. [CrossRef]
17. Yu, W. Chapter 8—PID Admittance Control in Task Space. In *PID Control with Intelligent Compensation for Exoskeleton Robots*; Yu, W., Ed.; Academic Press: Cambridge, MA, USA, 2018; pp. 139–158. [CrossRef]
18. Chung, J.-C.; Huang, C.-H.; Chou, H.-C.; Kuo, C.-H. Trajectory Planning of a Redundant Manipulator from Investigations of Upper Limb Motions of Human Beings. *IFAC Proc. Vol.* **2012**, *45*, 538–543. [CrossRef]
19. Laitenberger, M.; Raison, M.; Perie, D.; Begon, M. Refinement of the upper limb joint kinematics and dynamics using a subject-specific closed-loop forearm model. *Multibody Syst. Dyn.* **2014**, *33*. [CrossRef]
20. Mao, Y.; Agrawal, S.K. Wearable Cable-driven Upper Arm Exoskeleton—Motion with Transmitted Joint Force and Moment Minimization. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation, Anchorage, AK, USA, 3–7 May 2010; IEEE: New York, NY, USA, 2010; pp. 4334–4339. [CrossRef]
21. Ballesteros-Escamilla, M.; Cruz-Ortiz, D.; Salgado, I.; Chairez, I. Hybrid position/force output feedback second-order sliding mode control for a prototype of an active orthosis used in back-assisted mobilization. *Med Biol. Eng. Comput.* **2019**, *57*, 1843–1860. [CrossRef] [PubMed]
22. Wang, H.; Xu, H.; Tian, Y.; Tang, H. α-Variable adaptive model free control of iReHave upper-limb exoskeleton. *Adv. Eng. Softw.* **2020**, *148*, 102872. [CrossRef]

23. Cohen, A.; Or, Y. Modeling the dynamics and control of rehabilitative exoskeleton with robotic crutches. *Int. J. Adv. Robot. Syst.* **2018**, *15*, 1729881418761137. [CrossRef]
24. Ghobadi, M.; Sosnoff, J.; Kesavadas, T.; Esfahani, E.T. Using Mini Minimum Jerk Model for Human Activity Classification in Home-Based Monitoring. In Proceedings of the IEEE/Ras-Embs International Conference on Rehabilitation Robotics, Singapore, 11–14 August 2015; Yu, H.Y., Braun, D., Campolo, D., Eds.; IEEE: New York, NY, USA, 2015; pp. 909–912.
25. Wang, C.; Peng, L.; Hou, Z.G.; Li, J.; Luo, L.; Chen, S.; Wang, W. Kinematic Redundancy Analysis during Goal-Directed Motion for Trajectory Planning of an Upper-Limb Exoskeleton Robot. In Proceedings of the 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Berlin, Germany, 23-27 July 2019; IEEE: New York, NY, USA, 2019; Volume 2019, pp. 5251–5255. [CrossRef]
26. Kim, H.; Miller, L.M.; Byl, N.; Abrams, G.M.; Rosen, J. Redundancy resolution of the human arm and an upper limb exoskeleton. *IEEE Trans. Bio-Med Eng.* **2012**, *59*, 1770–1779. [CrossRef]
27. Soltani-Zarrin, R.; Zeiaee, A.; Langari, R.; Robson, N. Reference path generation for upper-arm exoskeletons considering scapulohumeral rhythms. In Proceedings of the 2017 International Conference on Rehabilitation Robotics (ICORR), London, UK, 17–20 July 2017; pp. 753–758.
28. Xie, Q.; Deng, Z.; Yu, H.; Meng, Q. A General Kinematics Model for Trajectory Planning of Upper Limb Exoskeleton Robots. In Proceedings of the Intelligent Robotics and Applications: 12th International Conference, ICIRA 2019, Shenyang, China, 8–11 August 2019; p. 63.
29. Guo, X.D.; Wang, Y.; Tong, H.T. *Pose Control of End-Effector of 6-Dofs Exoskeleton Arm for Functional Rehabilitation*; World Scientific Publ. Co Pte Ltd.: Singapore, 2016; pp. 1105–1113.
30. Głowiński, S.; Krzyżyński, T.; Pecolt, S.; Maciejewski, I. Design of motion trajectory of an arm exoskeleton. *Arch. Appl. Mech.* **2015**, *85*, 75–87. [CrossRef]
31. Friedman, J.; Flash, T. Trajectory of the index finger during grasping. *Exp. Brain Res.* **2009**, *196*, 497–509. [CrossRef]
32. Flash, T.; Meirovitch, Y.; Barliya, A. Models of human movement: Trajectory planning and inverse kinematics studies. *Robot. Auton. Syst.* **2013**, *61*, 330–339. [CrossRef]
33. Ronsse, R.; Lenzi, T.; Vitiello, N.; Koopman, B.; van Asseldonk, E.; De Rossi, S.M.; van den Kieboom, J.; van der Kooij, H.; Carrozza, M.C.; Ijspeert, A.J. Oscillator-based assistance of cyclical movements: Model-based and model-free approaches. *Med. Biol. Eng. Comput.* **2011**, *49*, 1173–1185. [CrossRef]
34. Meng, Q.Y.; Tan, S.L.; Yu, H.L.; Meng, Q.L.; Fang, Y.F. Trajectory planning and realizing of an exoskeleton device for hand rehabilitation based on sEMG control. In *Applied Mechanics and Materials*; Trans Tech Publications Ltd.: Stafa-Zurich, Switzerland, 2014; pp. 1015–1020.
35. Hogan, N.; Winters, J.M. Principles underlying movement organization: Upper limb. In *Multiple Muscle Systems*; Springer: New York, NY, USA, 1990; pp. 182–194.
36. Biess, A.; Nagurka, M.; Flash, T. Simulating discrete and rhythmic multi-joint human arm movements by optimization of nonlinear performance indices. *Biol. Cybern.* **2006**, *95*, 31–53. [CrossRef]
37. Li, Z.; Zuo, W.; Li, S. Zeroing dynamics method for motion control of industrial upper-limb exoskeleton system with minimal potential energy modulation. *Measurement* **2020**, *163*, 107964. [CrossRef]
38. Rahman, M.M.; Choudhury, T.T.; Sidek, S.N.; Awang, A. Mathematical modeling and trajectory planning of hand finger movements. In Proceedings of the 2014 First Conference on Systems Informatics, Modelling and Simulation, Sheffield, UK, 29 April–1 May 2014; pp. 43–47.
39. Secco, E.L.; Visioli, A.; Magenes, G. Minimum jerk motion planning for a prosthetic finger. *J. Robot. Syst.* **2004**, *21*, 361–368. [CrossRef]
40. Zadravec, M.; Matjacic, Z. Development of optimization-based simulation tool for trajectory planning in planar arm reaching after stroke. In Proceedings of the 2012 4th IEEE Ras & Embs International Conference on Biomedical Robotics and Biomechatronics, Rome, Italy, 24–27 June 2012; Desai, J.P., Jay, L.P.S., Zollo, L., Eds.; IEEE: New York, NY, USA, 2012; pp. 1446–1450.
41. Zadravec, M.; Matjacic, Z. Planar arm movement trajectory formation: An optimization based simulation study. *Biocybern. Biomed. Eng.* **2013**, *33*, 106–117. [CrossRef]
42. Averta, G.; Della Santina, C.; Valenza, G.; Bicchi, A.; Bianchi, M. Exploiting upper-limb functional principal components for human-like motion generation of anthropomorphic robots. *J. Neuroeng. Rehabil.* **2020**, *17*, 63. [CrossRef] [PubMed]

43. Enya, T.; Yamane, M.; Nakamura, H.; Aoki, T.; Nishimoto, Y.; Yano, K.i. Upper limb flexion assistance based on minimum-jerk trajectory using wearable motion-assist robot. *IFAC Proc. Vol.* **2011**, *44*, 5962–5967. [CrossRef]
44. Pattacini, U.; Nori, F.; Natale, L.; Metta, G.; Sandini, G. An experimental evaluation of a novel minimum-jerk cartesian controller for humanoid robots. In Proceedings of the 2010 IEEE/RSJ international conference on intelligent robots and systems, Taipei, Taiwan, 18–22 October 2010; pp. 1668–1674.
45. Tang, S.; Chen, L.; Barsotti, M.; Hu, L.; Li, Y.; Wu, X.; Bai, L.; Frisoli, A.; Hou, W. Kinematic Synergy of Multi-DoF Movement in Upper Limb and Its Application for Rehabilitation Exoskeleton Motion Planning. *Front. Neurorobotics* **2019**, *13*. [CrossRef]
46. Taïx, M.; Tran, M.T.; Souères, P.; Guigon, E. Generating human-like reaching movements with a humanoid robot: A computational approach. *J. Comput. Sci.* **2013**, *4*, 269–284. [CrossRef]
47. Liu, K.; Xiong, C.-H.; He, L.; Chen, W.-B.; Huang, X.-L. Postural synergy based design of exoskeleton robot replicating human arm reaching movements. *Robot. Auton. Syst.* **2018**, *99*, 84–96. [CrossRef]
48. Vallery, H.; Van Asseldonk, E.H.; Buss, M.; Van Der Kooij, H. Reference trajectory generation for rehabilitation robots: Complementary limb motion estimation. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2008**, *17*, 23–30. [CrossRef]
49. Gong, S.Q.; Zhao, J.; Zhang, Z.Q.; Xie, B.Y. Task motion planning for anthropomorphic arms based on human arm movement primitives. *Ind. Robot.* **2020**, *13*. [CrossRef]
50. Santello, M.; Bianchi, M.; Gabiccini, M.; Ricciardi, E.; Salvietti, G.; Prattichizzo, D.; Ernst, M.; Moscatelli, A.; Jörntell, H.; Kappers, A.M. Hand synergies: Integration of robotics and neuroscience for understanding the control of biological and artificial hands. *Phys. Life Rev.* **2016**, *17*, 1–23. [CrossRef] [PubMed]
51. Frisoli, A.; Loconsole, C.; Bartalucci, R.; Bergamasco, M. A new bounded jerk on-line trajectory planning for mimicking human movements in robot-aided neurorehabilitation. *Robot. Auton. Syst.* **2013**, *61*, 404–415. [CrossRef]
52. Shahbazi, H.; Parandeh, R.; Jamshidi, K. Implementation of imitation learning using natural learner central pattern generator neural networks. *Neural Netw.* **2016**, *83*, 94–108. [CrossRef]
53. Billard, A. Learning motor skills by imitation: A biologically inspired robotic model. *Cybern. Syst.* **2001**, *32*, 155–193. [CrossRef]
54. Naghibi, S.S.; Fallah, A.; Maleki, A.; Ghassemi, F. Elbow angle generation during activities of daily living using a submovement prediction model. *Biol. Cybern.* **2020**, *114*, 389–402. [CrossRef]
55. Garrido, J.; Yu, W.; Li, X. Robot trajectory generation using modified hidden Markov model and Lloyd's algorithm in joint space. *Eng. Appl. Artif. Intell.* **2016**, *53*, 32–40. [CrossRef]
56. Deng, M.D.; Li, Z.J.; Kang, Y.; Chen, C.L.P.; Chu, X.L. A Learning-Based Hierarchical Control Scheme for an Exoskeleton Robot in Human-Robot Cooperative Manipulation. *IEEE Trans. Cybern.* **2020**, *50*, 112–125. [CrossRef]
57. Wasielica, M.; Wasik, M.; Kasinski, A.; Skrzypczynski, P. Interactive Programming of a Mechatronic System: A Small Humanoid Robot Example. In Proceedings of the 2013 IEEE/Asme International Conference on Advanced Intelligent Mechatronics, Wollongong, Australia, 9–12 July 2013; IEEE: New York, NY, USA, 2013; pp. 459–464.
58. Vasan, G.; Pilarski, P.M. Learning from Demonstration: Teaching a Myoelectric Prosthesis with an Intact Limb via Reinforcement Learning. In Proceedings of the 2017 International Conference on Rehabilitation Robotics, London, UK, 17–20 July 2017; Amirabdollahian, F., Burdet, E., Masia, L., Eds.; IEEE: New York, NY, USA, 2017; pp. 1457–1464.
59. Hernandez, S.; Raison, M.; Baron, L.; Achiche, S. Refinement of exoskeleton design using multibody modeling: An overview. In Proceedings of the CCToMM Mechanisms, Machines, and Mechatronics Symposium (M3), Ottawa, ON, Canada, 28–29 May 2015; p. 110.
60. Li, J.; Cao, Q.; Zhang, C.; Tao, C.; Ji, R. Position solution of a novel four-DOFs self-aligning exoskeleton mechanism for upper limb rehabilitation. *Mech. Mach. Theory* **2019**, *141*, 14–39. [CrossRef]
61. Li, J.; Li, S.; Zhang, L.; Tao, C.; Ji, R. Position solution and kinematic interference analysis of a novel parallel hip-assistive mechanism. *Mech. Mach. Theory* **2018**, *120*, 265–287. [CrossRef]
62. Lu, J.; Chen, W.; Tomizuka, M. Kinematic design and analysis of a 6-DOF upper limb exoskeleton model for a brain-machine interface study. *IFAC Proc. Vol.* **2013**, *46*, 293–300. [CrossRef]

63. Hasan, Z.; Thomas, J. Kinematic redundancy. In *Progress in Brain Research*; Elsevier: Amsterdam, The Netherlands, 1999; Volume 123, pp. 379–387.
64. Betts, J.T. Survey of numerical methods for trajectory optimization. *J. Guid. Control. Dyn.* **1998**, *21*, 193–207. [CrossRef]
65. Geoffroy, P.; Mansard, N.; Raison, M.; Achiche, S.; Todorov, E. From inverse kinematics to optimal control. In *Advances in Robot Kinematics*; Springer: Cham, Switzerland, 2014; pp. 409–418.
66. Cao, H.X.; Sun, S.Q.; Zhang, K.J.; Tang, Z.C. Visualized trajectory planning of flexible redundant robotic arm using a novel hybrid algorithm. *Optik* **2016**, *127*, 9974–9983. [CrossRef]
67. Maciejasz, P.; Eschweiler, J.; Gerlach-Hahn, K.; Jansen-Troy, A.; Leonhardt, S. A survey on robotic devices for upper limb rehabilitation. *J. Neuroeng. Rehabil.* **2014**, *11*, 3. [CrossRef] [PubMed]
68. Babaiasl, M.; Mahdioun, S.H.; Jaryani, P.; Yazdani, M. A review of technological and clinical aspects of robot-aided rehabilitation of upper-extremity after stroke. *Disabil. Rehabil. Assist. Technol.* **2016**, *11*, 263–280. [CrossRef] [PubMed]
69. Molteni, F.; Ballarati, R.; Guanziroli, E. Clinical Needs and Possible Perspectives in Rehabilitation Context. In *Mediterranean Conference on Medical and Biological Engineering and Computing—MEDICON 2019*; Springer: Cham, Switzerland, 2019; pp. 1714–1721.
70. Islam, M.R.; Spiewak, C.; Rahman, M.; Fareh, R. A Brief Review on Robotic Exoskeletons for Upper Extremity Rehabilitation to Find the Gap between Research Porotype and Commercial Type. *Adv. Robot. Autom.* **2017**, *6*. [CrossRef]
71. Asl, H.J.; Narikiyo, T.; Kawanishi, M. An assist-as-needed control scheme for robot-assisted rehabilitation. In Proceedings of the 2017 American Control Conference (ACC), Seattle, WA, USA, 24–26 May 2017; pp. 198–203.
72. Gull, M.; Bai, S.; Bak, T. A Review on Design of Upper Limb Exoskeletons. *Robotics* **2020**, *9*, 16. [CrossRef]
73. Perret, J.; Vercruysse, P. Advantages of Mechanical Backdrivability for Medical Applications of Force Control. In Proceedings of the Workshop on Computer/Robot Assisted Surgery (CRAS), Genoa, Italy, 14–16 October 2014.
74. Lee, D.J.; Bae, S.J.; Jang, S.H.; Chang, P.H. Design of a clinically relevant upper-limb exoskeleton robot for stroke patients with spasticity. In Proceedings of the 2017 International Conference on Rehabilitation Robotics (ICORR), London, UK, 17–20 July 2017; pp. 622–627.
75. Kim, J.Y.; Yang, U.J.; Park, K. Design, motion planning and control of frozen shoulder rehabilitation robot. *Int. J. Precis. Eng. Manuf.* **2014**, *15*, 1875–1881. [CrossRef]

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.