Mechatronics 68 (2020) 102367

Contents lists available at ScienceDirect

Mechatronics

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

A survey on modularity and distributivity in series-parallel hybrid robots☆

Shivesh Kumar[a,*], Hendrik Wöhrle[a,b], José de Gea Fernández[a], Andreas Müller[c], Frank Kirchner[a,d]

[a] Robotics Innovation Center, German Research Center for Artificial Intelligence (DFKI GmbH), Bremen 28359, Germany
[b] Intelligente autonome Sensor- und Aktor-Systeme, Fachhochschule Dortmund, Dortmund 44139, Germany
[c] Institute of Robotics, Johannes Kepler University, Linz 4040, Austria
[d] Universität Bremen, Fachbereich Mathematik und Informatik, Arbeitsgruppe Robotik, Bremen 28359, Germany

A R T I C L E I N F O

Keywords:
Series-parallel hybrid robots
Modular design
Distributed control
Parallel robots
Modeling and control

A B S T R A C T

Parallel mechanisms are used increasingly often as modular subsystem units in various robots and man-machine interfaces for their superior stiffness, payload-to-weight ratio and dynamic properties. This leads to series-parallel hybrid robotic systems which utilize closed loop linkages and parallel kinematic machines as an abstraction of a certain kinematic joint. This paper presents a survey of recently developed series-parallel hybrid robots in various application domains such as legged robotics, humanoids, exoskeletons and industrial automation. In particular, we focus on modular and distributive aspects of such systems with an intention to bring their current design paradigm into focus, which simplifies the robot development process by promoting the effective reuse of hardware and software components and overcomes the shortcomings of traditional serial robots like poor payload capacity and stiffness.

1. Introduction

Serial and tree-type robots are mechanical systems that are designed as a series of links connected by motor-actuated joints that extend from a base to a single or multiple end-effectors. They are well known for their versatility in applications, large workspace, simple modeling and control. Hence, they often represent the state-of-the-art in robotic systems. Typical examples of serial robot architectures include industrial robotic arms like UR-3/5/10/16 [1], KUKA LBR iiwa systems [2] etc. and examples of tree type robot include dual arm manipulators like Baxter robot by Rethink robotics [3], YuMi collaborative robot by ABB [4] etc. However, these types of robot architecture generally feature only limited precision, low structural stiffness, and poor dynamic characteristics. For these reasons, robots based on a pure tree type topology suffer from speed and torque limitations. On the other hand, a parallel robot is a mechanical system that uses several computer-controlled serial chains to support a single platform, or end-effector (EE). In contrast to serial robots, parallel devices can provide higher stiffness, speed, accuracy, and payload capacity. On the downside, they possess a reduced workspace and a more complex geometry which requires careful analysis and control. Parallel robots have been traditionally used in more tailored use cases such as fast pick-and-place applications [5], driving simulators [6], fast orientation devices [7] etc. A review on parallel robots and their applications can be found in Patel and George [8], Parallemic [9] and on parallel redundant robots can be found in Luces et al. [10]. A series-parallel hybrid robot is defined as a robot which is built from a serial or tree-type combination serial and parallel mechanisms. These robots combine the advantages of both serial and parallel architectures but also inherit their kinematic complexities. Fig. 1 shows schematics of exemplary serial, tree-type, parallel and series-parallel hybrid mechanisms.

Series-parallel hybrid designs combining the advantages of serial and parallel topologies are common in the field of heavy machinery, e.g., cranes [11], excavator arms [12], bulldozers like backhoe loader [13,14] etc. for a long time [15]. Such designs have also recently caught the attention of robotics researchers from industry and academia. For instance, the stiffness of an industrial manipulator can be significantly improved by including a simple parallelogram mechanism. In particular, industrial robots such as ABB's IRB4400, IRB6660, KUKA's KR 40-PA., KR 50-PA., KR 700-PA robots, and Comau's Smart NJ series, SR400 utilize this design concept [16,17]. In academics and R&D, the idea to use closed loop mechanisms and parallel kinematic machines (PKMs) has been utilized more liberally, giving rise to a number of biologically inspired lightweight robotic systems with good dynamic characteristics. Some prominent examples including Active Ankle

☆ This paper was recommended for publication by Associate Editor Cesare Fantuzzi.

* Corresponding author.
E-mail addresses: shivesh.kumar@dfki.de (S. Kumar), hendrik.woehrle@dfki.de (H. Wöhrle), jose.de_gea_fernandez@dfki.de (J. de Gea Fernández), a.mueller@jku.at (A. Müller), frank.kirchner@dfki.de (F. Kirchner).

https://doi.org/10.1016/j.mechatronics.2020.102367
Received 20 September 2018; Received in revised form 20 October 2019; Accepted 7 April 2020

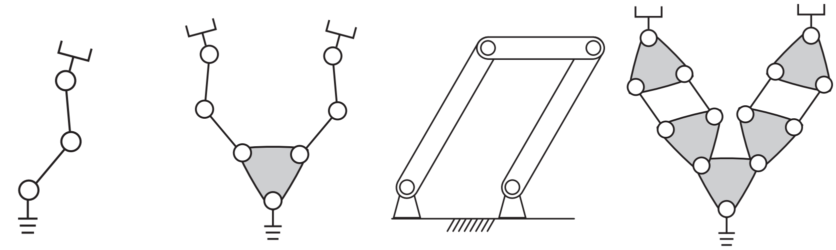


Fig. 1. Serial, tree-type, parallel and series-parallel hybrid mechanisms.

[18], Stewart platform, 2SPU + 1U[1] [20], double parallelogram linkage [21] etc. have been used in various hybrid robots like hominid CHARLIE [22], multi-legged robot MANTIS [23], RECUPERA full body exoskeleton [24], AXO-SUIT [25] as well as humanoid robots like LOLA [26] and THOR [27]. The design motivation of such hybrid robots is evident: use of PKM-based submechanisms helps designers to achieve lightweight, modular and compact design while enhancing the stiffness and dynamic characteristics of the robot.

Modular robotics has been an extensive area of research ever since the inception of first modular robot called CEBOT [28] (abbreviation of cellular robotic system) by Toshio Fukuda. This kind of robots provides a unique advantage over conventional robotic systems in terms of reusability, reconfigurability, and ease in manufacturing. Extensive surveys tracking the developments in this research field can be found in Yim et al. [29], Ahmadzadeh et al. [30] and Chennareddy et al. [31]. Modular robots can be classified based on their structure, locomotion, reconfigurability and form-factor. From a structural point of view, modular robots are classified into 5 categories: lattice, chain/tree, mobile, hybrid and truss. Series-parallel hybrid robots fall in the category of chain type modular robots with fixed configuration.[2] Although such designs were only sparsely reported in the past due to challenges in understanding their geometry and control, such problems are now being conquered by developments in modern kinematics such as screw theory and computational algebraic geometry. Hence, a number of series parallel hybrid robots for various application domains have been developed in the last decade which utilize PKM-based submechanism modules. Various aspects of such robots such as modularity and distributed control are similar to those of modular robots, but they are underrepresented in survey studies on modular robots. In fact, a systematic survey on series-parallel hybrid robots is missing in the literature. This paper intends to provide a survey on series-parallel hybrid robotic systems while emphasizing on their modular and distributive aspects which are central to a modular design paradigm.

Organization The paper is organized as follows: Section 2 provides a survey on various series-parallel hybrid robots in various application domains. Section 3 presents the notion of modularity and distributivity in these hybrid robots from a hardware and software perspective. Section 4 discusses the modeling and control methods for these robots while highlighting the theoretical challenges and adopted solution approaches. Section 5 presents a comparative study and discussion on the modular design paradigm which will help in building robots of the future by saving costs and integration time. It also points out the challenges and future research directions. Section 6 concludes the paper.

[1] In mechanism theory, it is typical to identify mechanisms using their type. For details, see [19].

[2] Reconfigurable parallel robots can change their motion mode according to task requirements. To the best knowledge of the authors, this concept has not yet been utilized in the design of hybrid robots.

2. Series-parallel hybrid designs

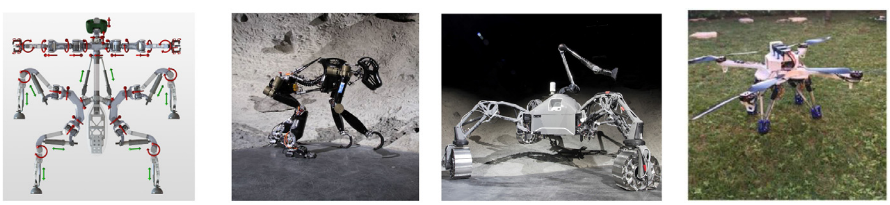
The survey presented in this section studies the evolution of series-parallel hybrid robots primarily in various domains: humanoids/bipeds, multi-legged robotic systems, exoskeletons or physical man-machine interfaces and industrial automation. Further, some overall trends in their evolution have been identified.

2.1. Humanoids

Humanoids are bipedal robots equipped with upper body, arms and head to mimic human anatomy. These are complex mechatronic systems with highly interdependent technical aspects. The last few decades of research in humanoid robotics have shown that humanoids designed for high dynamic performance require a stiff structure and good mass distribution [32]. Both of these properties can be achieved by utilizing parallel mechanisms in the design. Fig. 2a shows the bipedal robot LOLA [26] developed in 2006 by TU Munich, which is probably the first humanoid robot designed using a modular parallel joint concept. The non-linear transmission between the actuator space and output space in a parallel mechanism can be utilized advantageously by adjusting its design parameters to the torque-speed characteristics of typical movements of the robot [33]. Moreover, utilizing multi degrees of freedom (DOF) parallel mechanisms can also help in achieving a more uniform load distribution on the actuators. LOLA utilized a spatial slider crank mechanism in the knee joint and a two DOF rotational parallel mechanism in the ankle joint. Similarly, the robot AILA [34] revealed in 2010 by DFKI-RIC also used PKM modules for its wrist, neck and torso joints. The design of NASA VALKYRIE humanoid [35] as shown in Fig. 2b built by the NASA Johnson Space Center (JSC) to compete in the 2013 DARPA Robotics Challenge (DRC) Trials followed a similar design concept by utilizing PKM modules for its wrist, torso and ankle joints. The torque controlled humanoid TORO from DLR released in 2013 [36] and TALOS [37] from PAL Robotics are largely tree type systems but utilize a simple parallelogram linkage in its ankle for creating the pitch movement (see Fig. 2c). Humanoid robots THOR (2014) [27,38] and SAFFIR (2016) [39] from Virginia Tech. use 2 DOF PKM for hip roll-yaw and ankle joints and utilize Hoeken's mechanism for hip pitch and knee joints. LARMBOT [40] reported in 2016 is a humanoid robot prototype which utilizes two linearly actuated tripod parallel mechanisms as legs and a cable driven parallel mechanism (CPDR) for its torso design. CARL (reported in 2017) is a compliant walking leg using parallel redundant actuation which mimics the behavior of the human muscles [41]. RH5 is a lightweight and biologically inspired humanoid robot recently developed by DFKI-RIC in 2017 which uses linkages and PKM modules for most of its joints for e.g. hip flexion-extension, knee, ankle, torso and wrist [42] (see Fig. 2d). Disney Research in 2018 also reported a design of a bipedal robot [43] which has HYBRID LEGS [44] optimized for achieving a fast walking behavior.

(a) Lola [26] (TUM) (b) Valkyrie [35] (NASA) (c) TORO [36] (DLR) (d) RH5 (DFKI)

Fig. 2. Humanoid robots with series-parallel hybrid design.



(a) Mantis (DFKI) (b) Charlie (DFKI) (c) SherpaTT (DFKI) (d) HeritageBot [48]

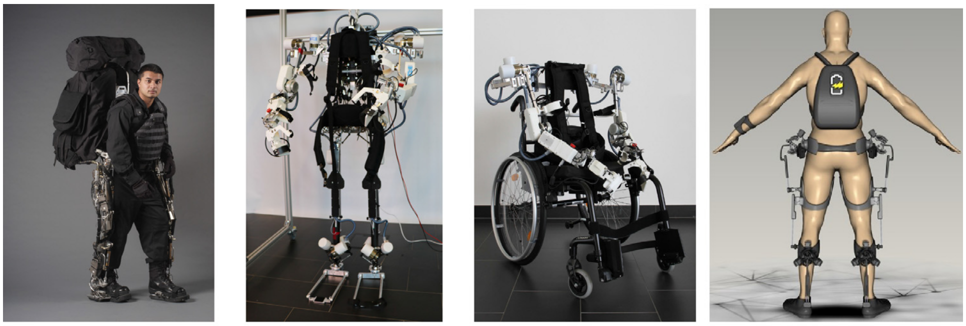
Fig. 3. Multi-legged robots with series-parallel hybrid design.

2.2. Multi-legged robots

Closed loop linkages and parallel mechanisms are increasingly often used in multi-legged robots which are specifically designed for high-payload applications. With their use, certain joints can be strengthened without compromising the overall leg inertia. Fig. 3b shows the hominid robot Charlie [22] developed in 2013, featuring a Stewart platform of type 6-RUS as a six DOF active joint module in spine and neck. It also utilizes another parallel mechanism in the ankle joint. This robot supports both bipedal and quadrupedal walking gaits. The multi-legged robot Mantis [23] developed in 2016 contains PKMs of type 2-SPU + 1U [20] in its ankle joints and slider-crank mechanisms that drive certain revolute joints in its legs and torso (see Fig. 3a). The SherpaTT rover (2016), which is an active suspension wheel-leg hybrid robot, uses a closed loop linkage in inner and outer leg joints [45,46]. The system weighs 166 kg in total, can carry a payload of upto 80 kg and has a variable footprint size ranging from 1 m^2 (1m × 1 m) to 6.76 m^2 (2.4 m × 2.4 m) and the body height can be adjusted in the range of 0.8 m to 1.8 m. ATHLETE (first constructed in 2005) [47] is an active suspension rover with purely tree type architecture with its legs mounted on a hexagonal frame 2.75 m wide and can achieve a maximum height of 2 m. Its total weight is 850 kg and has a payload capacity of 300 kg. When compared with ATHLETE, SherpaTT rover demonstrates better payload to weight ratio which is directly a result of its series-parallel hybrid leg architecture. HeritageBot (2018) [48] from University of Cassino is a hybrid robot which is capable of flying and legged locomotion on ground. Its modular design makes use of four tripod parallel mechanisms for the leg design. Recently in 2017 [49] and 2019 [50], there have been attempts to automate a walking excavator called Menzi Muck M545 which has been manually controlled in the past. These kind of excavators are not traditionally seen as robots, but in future their automation will put them in the category wheel-leg series-parallel hybrid robot. MIT Cheetah (2017) [51] developed for high speed running utilizes a four bar linkage in its knee joint in order to minimize the overall leg inertia. In 2019, a quadrupedal research platform Stoch [52] was reported which also includes a parallelogram linkage in the knee joint.

2.3. Exoskeletons

In exoskeletons or physical man-machine interfaces, most joints require a limited range of motion because most of the human joints like the wrist or ankle are not able to perform a 360° rotation movement. Hence, in exoskeletons based on serial kinematic chain, a physical limitation of joint movements is necessary for safety reasons. Software based joint limits may fail and, hence, additional mechanical end stops are required at each joint. The use of parallel mechanisms in exoskeletons can not only lead to a lightweight design but also their limited workspace may be exploited as an additional safety feature. An additional benefit of using parallel mechanisms is that they allow motion about a remote center of rotation which is especially useful in the design of exoskeletons. This provides better opportunities for the designers to mount the mechanism around the complex human anatomy. Some early exoskeleton designs utilizing closed loop linkages are Bleex [53] as shown in Fig. 4a and ExoHiker [54] were revealed in 2004 and 2005 respectively by the Berkeley Robotics & Human Engineering Laboratory. Fig. 4b demonstrates a highly modular light weight Recupera full-body exoskeleton [24,55] with 32 active DOFs which is built by combining several higher DOF joint modules: a Stewart platform of type 6-UPS in torso, a double parallelogram [21] in shoulder for flexion–extension movement and Active Ankle mechanism [18] as a 3 DOF joint in hip and ankle. Due to high modularity of its mechanics and electronics design, the upper body including the two arms can be mounted on a wheelchair for rehabilitation applications (see Fig. 4c). The Recupera was first conceptualized in 2016 and the system was fully integrated and tested in 2018. Two related parallel developments reported in 2017 have been the design of AXO-SUIT [25] which utilizes a passive double parallelogram mechanism in the shoulder joint and Harmony dual arm exoskeleton [56] which employs a parallelogram mechanism in the shoulder girdle joint. Another recent development in 2017 is the Spherical Exo Suit as shown in Fig. 4d which employs Agile Wrist mechanism as a 3 DOF spherical joint module at hip and ankle joints [57].



(a) Berkeley Lower Extremity Exo [53] (UCB)
(b) Recupera-Reha full body exoskeleton (DFKI)
(c) Recupera-Reha Wheelchair Exoskeleton (DFKI)
(d) Spherical Exoskeleton Suit (SFU, Canada)

Fig. 4. Exoskeletons with series-parallel hybrid design.

(a) Logabox LX4 [58]
(b) ABB IRB4400 (Courtesy: ABB)
(c) 6-axis DELTA robot (Courtesy: Fanuc)

Fig. 5. Some series-parallel hybrid robots used in industrial automation.

2.4. Industrial automation

An early example of series parallel hybrid robot is Logabex's LX4 robot [59] (reported in 1990) shown in Fig. 5a which piles four Stewart platform modules in series to achieve a redundant series-parallel hybrid manipulator with large workspace and good payload (75 kg) to weight (120 kg) ratio. The control of such a robot is difficult [58]. Nevertheless, these kind of robots are also gaining recognition in the industrial applications. For high stiffness applications, serial robot designs are often complemented by a parallelogram mechanism in the design (see Fig. 5b). Industrial robots such as ABB's IRB4400, IRB6660, KUKA's KR 40-PA, KR 50-PA, KR 700-PA robots, and Comau's Smart NJ series, SR400 utilize parallelograms for increasing stiffness of one or several joints To and Webb [16], Gautier et al. [17]. The 3-DOF DELTA robot has gained high popularity for fast pick and place operations in the industry. When the original patents for the Delta robot expired in 2006 (Europe) and 2007 (North America), the number of competitors in this market increased rapidly. Due to its increased adoption, a demand for mounting an active wrist was realized and several such robots with 1, 2 and 3 DOF wrists were developed. A comparative study of various series-parallel delta robot designs can be found in Brinker et al. [60]. FANUC M-3iA/6A Delta Robot is an interesting six axis series-parallel hybrid robot designed for pick and place operations in the food industry (see Fig. 5c). It is also available in 4 axis version with a single axis wrist design.

2.5. Overall trends

From the above survey including 31 hybrid robot designs, it can be seen that the adoption of series-parallel hybrid robots has indeed become evident in the last two decades in various application areas such as humanoids/bipeds, multi-legged robotic systems, exoskeletons or physical man-machine interfaces and industrial automation. Out of these domains, the most dominant application of such a design architecture is in the area of legged locomotion (58% of the robots) which includes humanoids (35%) and multi-legged systems (23%) since it can provide opportunities for enhancing the stiffness of the robot, optimizing the mass distribution in the robot design and bringing the mechanical advantage of the non-linear transmission. Further, achieving a large workspace is not a priority in this application domain. There is a growing adoption of this design architecture in the area of exoskeletons (23%) especially in the last 5 years. Further, many series-parallel hybrid robots can be found in the field of industrial automation (13%) however the variation in their design seems to be rather conservative. Some applications of hybrid robots can also be found in the domain of underwater robotics (6%) for example, ORION7P manipulator [61], Autonomous Underwater Vehicle (AUV) LENG [62] etc. Fig. 6 shows the distribution of series-parallel hybrid robot designs considered in this survey in different application areas.

Further, it is also interesting to study the extent to which parallel mechanisms are used in the design of series-parallel hybrid robots. Table 1 shows the number of parallel submechanism modules of different complexities (ranging from 1 DOF linkages to 6 DOF PKMs) used in various series-parallel hybrid robots from the top three application areas. It can be noticed that lower DOF parallel mechanisms (with 1 or 2 DOF) are more often used in the design of hybrid robots in comparison to the higher DOF parallel mechanisms. Nevertheless, there are some hybrid robots (e.g. CHARLIE, RECUPERA Exoskeleton etc) which feature those mechanisms. The table also notes the total number of DOFs that are parallel actuated against the total DOF of the system. The percentage of parallel actuated DOFs over the total DOF of various robots are plotted against their development year in Fig. 7. It can be observed that

Table 1
Overview of number of parallel submechanism based modules with different complexities in series-parallel hybrid robots.

Robot name (year)	Parallel mechanism (number of modules)					Composition	
	1-DOF	2-DOF	3-DOF	4-DOF	6-DOF	Total DOFs	Parallel DOFs
LOLA (2006)	2	2	–	–	–	25	6
AILA (2010)	2	2	–	–	–	20	6
Valkyrie (2013)	–	5	–	–	–	35	10
TORO (2014)	2	–	–	–	–	27	2
THOR (2014)	4	4	–	–	–	30	12
SAFFIR (2016)	4	4	–	–	–	30	12
LARMBOT (2016)	–	–	2	1	–	22	10
TALOS (2017)	2	–	–	–	–	27	2
RH5 (2017)	5	5	–	–	–	32	15
Disney Biped (2018)	–	–	–	–	2	12	12
Charlie (2013)	–	2	–	–	2	36	16
MANTIS (2016)	6	4	–	–	2	45	14
SherpaTT (2016)	8	–	–	–	–	20	8
HeritageBOT (2016)	–	–	4	–	–	16	12
MIT Cheetah (2017)	4	–	–	–	–	12	4
Stoch (2019)	4	–	–	–	–	12	4
BLEEX (2004)	8	–	–	–	–	14	8
Recupera Wheelchair Exo (2016)	2	–	–	–	–	10	2
Harmony (2017)	4	–	–	–	–	14	4
AXO-SUIT (2017)	2	–	–	–	–	15	2
SPHERICAL EXO SUIT (2017)	–	–	4	–	–	12	12
Recupera Full Body Exo (2018)	2	–	4	–	1	30	20

Table 2
Linkages and PKM modules in series-parallel hybrid robots where platform coordinates can be measured directly. (red, green and yellow colors denote active, EE and other passive joints).

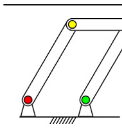
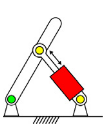
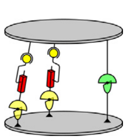
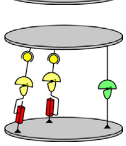
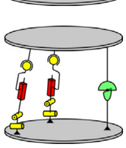
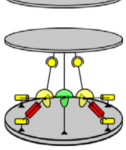
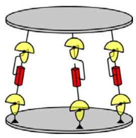
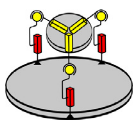
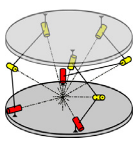
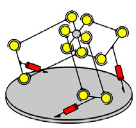
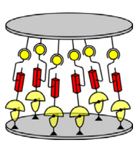
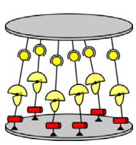
Mechanism			Practical applications	
Schematic	Type	Mobility (m)	Application	Hybrid robot
	1-RRRR	1 DOF rotational	Ankle pitch Series-elastic leg Ankle pitch Shoulder flexion-extension Joint 2 and/or Joint 3 Knee joint	TORO humanoid [36] ATRIAS [63] TALOS humanoid [37]Recupera-Reha [21], Harmony [56]ABB IRB4400, KUKA KR 40-PA etc. MIT Cheetah [51], Stoch [52]
	1-RRPR	1 DOF rotational	Hip, Torso Hip flexion-extension, Knee Inner and Outer leg joints Hip-Knee, Knee-Ankle joints Shoulder	Mantis [23] RH5 [42], HADE leg [64] SherpaTT rover [46] CARL [65] Orion7P [61]
	2-SPU+1U	2 DOF universal	Wrist, Torso Ankle Ankle Hip Roll-Yaw, Ankle Thurster steering	RH5 humanoid [42] Mantis [23] Lola humanoid [26] THOR [27], SAFFiR [39] AUV Leng, EurEx [62]
	2-PUS+1U	2 DOF universal	Wrist, Ankle and Torso	Valkyrie humanoid [35]
	2-SPRR+1U	2 DOF universal	Ankle	RH5 [66]
	2-SU[1-RRPR]+1U	2 DOF universal	Ankle	Charlie [22]

Table 3
PKM modules in series-parallel hybrid robots where platform coordinates can not be sensed. (red and yellow colors denote active and passive joints respectively).

Mechanism			Practical applications	
Schematic	Type	Mobility (m)	Application	Hybrid robot
	3-UPU	3 DOF translational	Leg	LARMBOT [40], HERITAGEBOT [48]
	3-PSP	3 DOF 2 Rotational 1 Translation	Torso [71], Wrist [72]	R1 humanoid [67]
	3-RRR	3 DOF spherical	Hip, Ankle	SPHERICAL EXO SUIT [57]
	3-R[2-SS]	3 DOF almost spherical	Hip, Ankle	RECUPERA-REHA full-body exo [69]
	6-UPS	6 DOF free	Torso	RECUPERA-REHA full-body exo [24]
	6-RUS	6 DOF free	Torso, Neck	CHARLIE homonid [22]

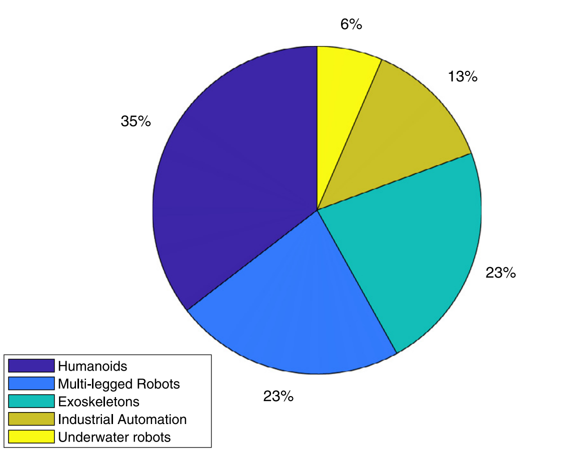


Fig. 6. Distribution of 31 series-parallel hybrid robot designs in different application areas.

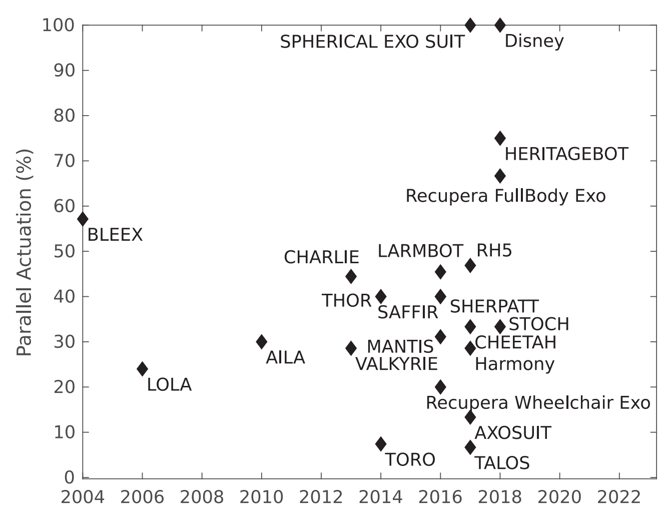


Fig. 7. Percentage of parallel actuation in various robots against their development year.

the recent parallel hybrid robots have more parallel mechanisms based actuation in comparison to the older systems. Fig. 7 also shows that a large number of hybrid designs have been reported in the past 5 years which demonstrates the growing popularity of this architecture.

Lastly, looking at the complexity of parallel mechanisms used in such designs, two observations can be made: (1) the recent hybrid robot designs contain more integrated higher DOF parallel mechanisms in comparison with the older hybrid robots, (2) use of complex parallel mechanisms is more prevalent in academia and research in comparison to the industry. Overall, one can conclude that there is a positive trend in the use of series-parallel hybrid designs and the complexity of these systems has been steadily increasing.

3. Modular and distributive aspects in design

In this section, we present the notion of modularity and distributivity in the series-parallel hybrid robots from a hardware and software perspective. The subsection Hardware is further subdivided into Mechanics and Electronics design. The subsection Software discusses the software architectures used in the control of such robotic systems. The goal of this section is to present the most popular architectures used in the mechatronic system design of series-parallel hybrid robots.

3.1. Hardware

3.1.1. Mechanics

Hybrid robots are robots that are composed of a series of serial or parallel submechanisms. From the perspective of mechanical design, it is interesting to study the types of parallel mechanisms that are repeatedly used in the design of hybrid robots that were studied from an application perspective in Section 2. Tables 2 and 3 present a list of closed loop linkages and parallel submechanism modules which have been utilized in series-parallel hybrid robot designs discussed in the previous section. In all cases, it can be observed that these submechanisms are used as a mechanical generator of m-dimensional motion subspaces of $SE(3)$, i.e., they serve as an abstraction to either an active lower pair joint (for e.g. revolute joint, spherical joint etc.) or universal joint which are building blocks of most robotic systems. It can be immediately noticed from these tables that parallel submechanism modules are mostly used as abstractions to orientational joints, exceptions are their application as 6 DOF joint in Charlie, Recupera-Reha exoskeleton and 2R1T wrist in R1 humanoid [67]. The most popular abstractions are discussed in the following.

- 1 DOF Revolute joint: Abstraction of a revolute joint is basically done using variants of a four bar linkage. For a simple 1: 1 transmission, a parallelogram [36,37] or double-parallelogram linkage [21] is often employed. In comparison to direct drive joints, the main motivation here is to reduce the resulting inertia of the robot or to create a virtual center of rotation. The slider-crank mechanism (identified as 1-RR$\underline{\text{P}}$R in Table 2) is used to exploit the non-linear transmission properties of the mechanism. Also, the prismatic actuation gives the possibility to mount the actuator along the link longitudnally which is advantageous from a construction perspective.
- 2 DOF Universal joint: Abstraction of a universal joint is very useful when the joint requires only a limited range of rotational motion and is to be placed away from the base of the robot. Hence, the most common application is the design of ankle joint in humanoids or legged robots. They also have been used in the design of wrist and torso mechanisms. All orientational parallel manipulators that have been used in this context are equipped with a passive leg containing the universal joint [68]. Such designs are very advantageous because they do not have output singularities and provide good stiffness to the moving platform. Also, it is easy to install rotary encoders to measure the orientation of the platform directly so that forward kinematics of the mechanism need not be solved in real time.
- 3 DOF Spherical joint: Since the workspace of 3 DOF orientational parallel manipulators is limited, they have been mostly used in the design of exoskeletons for the abstraction of hip and ankle joints. Agile Eye [7], which was originally developed as a fast camera orienting device, is used as hip and ankle module in Spherical Exo Suit concept. The disadvantage of this design is that it requires all the revolute joint axes to intersect at one point, which is not good for high payload or impact applications. Active Ankle [69] is an interesting parallel mechanism which overcomes this problem by utilizing spatial quadilateral mechanism in each leg where only push-pull forces can exist in each leg. However, it is an almost spherical mechanism and is only suited for applications where small translation of the EE point can be tolerated.
- Six DOF free joint: Many joints in humans like spine are actually very complicated and not easy to abstract using lower kinematic pairs. To provide a 6 DOF active spine to human-like robots, variants of Stewart Gough platform have been utilized. In our survey, we could find designs with both rotary (6-$\underline{\text{R}}$US) and prismatic actuation (6-U$\underline{\text{P}}$S). The active spine can also be used as a 6 DOF force-torque sensor and improves the workspace of limbs and lowers the velocity requirements of other limb joints [70]. The disadvantage of using such mechanisms is the complicated nature of their forward kinematic problem.

Table 2 shows the list of linkages and parallel submechanisms where the platform coordinates (highlighted with green) are a subset of coordinates used to describe the mechanism. Since, it is well known that it's difficult to solve the forward kinematics of parallel robots in real time, there is a tendency to choose parallel mechanisms where additional sensors can be integrated to measure the platform coordinates. Table 3 shows a list of parallel submechanisms where the platform coordinates are not a subset of mechanism's coordinates. Here, its also not possible to put extra sensors to measure the platform coordinates directly but in some cases they may be integrated in other passive joints to simplify the calculation of platform coordinates from actuator states. Hence, the use of such parallel submechanisms is less common in the design of series-parallel hybrid robots in comparison to the ones presented in Table 2.

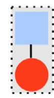
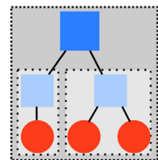
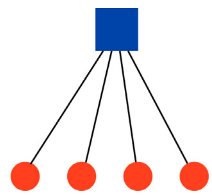
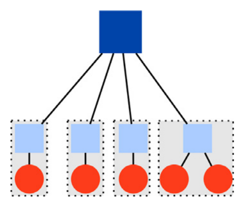
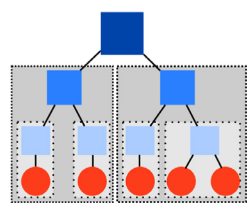
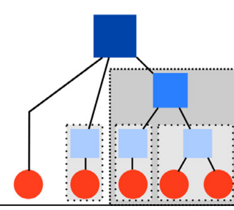
Overall, two observations can be made from this survey: submechanism modules are used as a mechanical generator of a motion subspace (revolute, universal, spherical, free joint etc.) and the same type of submechanism with different physical parameters is utilized as a module to serve different purposes (ankle, wrist, torso joints etc.) in the same robot. These observations clearly demonstrate the modular nature of the design of series-parallel hybrid robots in the mechanical domain.

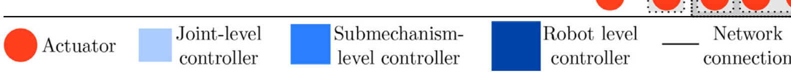
3.1.2. Electronics

Series-parallel robots demand certain actuators to work together in order to perform the desired motion corresponding to a certain kinematic joint. This requires local exchange of data in between them in order to implement their control. One important development in robot electronics that is especially helpful for series-parallel hybrid robots are decentralized or distributed processing and control architectures. This section presents a study of different electronics architectures that are used in such systems and how they support their modular design and distributed control.

A hierarchy with five types of modularity can be defined as shown in Table 4. A prerequisite for distributed architectures is to have actuator control units (ACUs) (Table 4(a)), i.e., actuators that are controlled by local controllers. They usually contain highly integrated, fully operational modules that include an actuator, sensors, and electronics for signal processing, communication and control. Several such ACUs can be assmebed to buld subsystem level mechanisms or submechanisms (Table 4(b)). The control of multiple joints in a synchronized fashion (see Sections 3.2 and 4.4) is especially important for the control of parallel and, hence, hybrid systems. In classical (i.e., centralized) control systems, all actuators are connected to one central host computing system (Table 4(c)). In contrast to these, distributed processing systems (Table 4(d, e)) consist of multiple computing devices (often called *processing elements* or *nodes*) that are connected via a network to exchange data with one another. In certain cases it might also occur that different combinations of electronic architectures are used, e.g., actuators, joint modules, and submechanism modules connected to a central host (Table 4(f)). In both centralized and distributed systems, a communication network is required to exchange data between sensors, actuators, and controllers. This is accomplished by using field-bus systems that are also often used in the industrial or automotive fields like CAN, CAN-FD or higher-level protocols like PROFIBUS [83]. Nowadays, different modified Ethernet-based protocols with realtime support [84] are becoming

Table 4
Types of electronic architectures for actuator, mechanism and robot control.

Architecture		Examples
Type	Schematic	
(a) Actuator control unit		ANYdrive [73] Dynamixel [74] Dynamixel Pro [75] Kinova KA-58 and KA-75+ actuators [76] Schunk PowerCubeTree actuators [77] KIT Sensor-Actuator-Controller Unit [78]
(b) Submechanism		Robonaut hand [79] ANYpulator [80] Charlie spine [22]
(c) Centralized		LARMbot [40], HeritageBot [48]
(d) Central host + joint modules		TORO [36] ANYmal [81] Recupera-Reha full-body exo [55]
(e) Central host + submechanism modules		ECCE humanoid robot [82]
(f) Combinations of centralized, joint and submechanism modules		Charlie hominid [22], RH5



increasingly poplar, for example PROFINET [85], EtherCAT [86] or SERCOS [87]. Although the required bandwidth to coordinate several joints is quite low (depending on the control strategy, the exchanged data consists of simple parameters, e.g., desired position, velocity, torque), the response time should be as short as possible. The nodes contribute to the overall processing of the system by performing specific processing tasks and providing additional capabilities, for example peripherals. Distributed architectures are important for modularity, since they provide the possibility to build independent substructures of robot systems. Local controllers implemented in the distributed hardware can be used with high sensor sampling rates and a low delay in comparison to a centralized approach.

Commercial examples for ACUs are the Robotis Dynamixel and Dynamixel Pro series [74,75], the Kinova KA-58 and KA-75+ actuators [76], the Schunk PowerCube, PR 2, PRH, PDU 2 and PSM 2 series, Tree actuators [77] or the ANYdrive [73]. However, reusable joint modules are not only used commercially, but also developed by research facilities to build novel and experimental robotic systems, such as the DLR LWR III robot [88] or DFKI Charlie [89] joints. Further examples support often additional features, such as joint-modules with series elastic actuation as used in the Walk-Man [90] and RoboSimian [91] robots, an interated inertial measurement unit [78], or even space compliance such as the DFKI-X joint [92]. However, it is important to note that many research joint modules usually don't have integrated, but closely attached electronic controllers. Series-parallel hybrid robots that contain reusable joint modules are the Charlie [22], Mantis [23] and Recupera [24] systems. In these examples, one or a few actuators are combined with a processing system that is responsible for the first level control of an actuator. Joint-specific limits (position, velocity, torque) can also be implemented in the joint-level controller for a hierarchical safety implementation. Joint level modules can be used to create systems with (2) submechanism-level modularity, i.e., more complex subsystems that need to be controlled as a whole. Examples are classical serial mechanisms like limbs or parallel mechanisms as shown in Tab. 2 and 3. Examples for such independently working modules are limbs in the ECCE humanoid robot [82] or the Robonaut hand [79] or the spine in Char-

Table 5
Robot middleware, mechanism of distribution and supported architectures.

Name	Distribution mechanism/pattern	Known supported architectures
ROS [97]	Proprietary publish/subscribe	$\times 86/\times 64$, ARM w. MMU
Yarp [98]	Proprietary *Observer pattern*-based	$\times 86/\times 64$, ARM w. MMU
Orocos [100]	CORBA/Message Queues	$\times 86/\times 64$, ARM w. MMU
RoCK [102]	Based on Orocos	$\times 86/\times 64$, ARM w. MMU
FINROC [104]	Peer to peer, publish/subscribe	$\times 86/\times 64$
ROS2 [106]	DDS	$\times 86/\times 64$, ARM w. MMU

LIE [22]. One important point to note is that the use of these mechanisms requires to solve the (forward/inverse) kinematics and dynamics of the corresponding subsystem, which can be performed locally to maintain modularity. This constitutes the second-level control of the robotic system. The next modularity level is (3) the complete robot system itself (which can in turn build teams or swarms with other robots). Regarding the processing architectures, it can be observed that even in robots that consist of independent modules, the distributed computing located in the modules are usually connected to a central host that is responsible for overall control. Different devices can be used to implement a distributed robotic control system, e.g., microcontrollers as in ECCE [82] or FPGA/processor combinations as in the ROBONAUT [93]/ROBONAUT 2 [94] and VALKYRIE [35] robots or CHARLIE [22], MANTIS [23] and RECUPERA [24].

3.2. Software

In modular robots with distributed control it is important to enable different components to exchange data in order to synchronize the control of several joints. This is especially important for modular robots with series-parallel hybrid architectures. In this case, the overall operation of the robot is distributed across multiple heterogeneous processors to maintain modularity and, hence, several actuators have to be coordinated. Accordingly, some sort of communication middleware is required [95,96]. The middleware should provide a coherent interface to the different hardware components and provide mechanisms of data transfer. This section provides an overview of different middleware frameworks popular in the field of robotics and their shortcomings with regard to their application in hybrid robots, some newly evolving middleware which try to address these shortcomings and lastly some perspective on the importance of a proper fieldbus system.

Many different robot middleware frameworks are available in for the research community, see Table 5 for an overview. The most popular examples probably the Robot Operating System (ROS) [97] or Yet Another Robot Platform (YARP) [98]. Each framework has specific advantages and core application areas (e.g., perception, manipulation, locomotion for YARP, and navigation, planning for ROS [99]). All of these frameworks support some kind of modularity. However, the approaches to establish communication channels between modules or even to provide middleware functionality are very diverse. In ROS, for example, the functionality is implemented inside so-called *nodes*. Each node runs inside a different process, the communication between nodes is mediated by an anonymous publish-subscribe middleware. However, for modular hybrid robotic systems, ROS has a number of shortcomings. ROS, for instance, has neither real time capabilities nor is it possible to use ROS on small *bare metal* systems like microcontrollers in actuator modules (see Section 3.1.2). Finally, ROS does not support fieldbus or embedded communication systems like Ethercat or even CAN directly. Hence, it is not possible to, e.g., instantiate ROS nodes that run on actuator modules to control a specific robot joint directly or even build a complete system out of them. Similar limitations are prevalent in most other robot frameworks: it is possible to build modular distributed systems on a software level, but not on a bare-metal hardware level. Synchronization of different actuators and, hence, real-time capabilities, are of high importance for parallel and hybrid systems. A limited support of these capabilities are offered by the realtime variant of Orocos, Orocos RT [100,101] and, hence, the derived Robot Construction Kit (RoCK) [102]. However, similar to ROS, these frameworks do not support low-level systems. This is also the case for the other robot frameworks shown in Table 5, which support $\times 86/\times 64$ architectures and ARM processors that contain a memory management unit (MMU), since they require a (non-embedded) operating system.

As a consequence, one aim of recent developments is to support baremetal controllers and support fieldbuses. Following this approach will also enhance the submechanism level modularity in series-parallel hybrid robots, since in this case multiple joints need to be controlled in combination (see Section 3.1.2). Examples where this design principle has been used are the computation of the inverse kinematics of a Stewart platform representing the spine in the chimp-like robot Charlie on a softcore CPU in CHARLIE [22]) to satisfy constraints regarding size and power consumption which was built using NDLCom to communicate between the controllers [103]. Another recent example is Finroc [104], which has been used to, e.g., implement a distributed controller for a series-parallel hybrid leg with redundant actuation [105]. Nevertheless, with increasing requirements and the need to use small embedded systems, new robot software frameworks like ROS2 [106], are being developed. They support technologies like Protocol Buffers [107], ZeroMQ [108] and the Data Distribution Service (DDS) [109,110] and might satisfy the constraints of mechanism-level modularity in future applications. Although ROS2 uses the DDS, which does not result in a performance benefit of ROS 2 over the plain TCP or UDP used in ROS [111], it supports realtime requirements and guarantees different Quality of Service (QoS) levels. Another important development that has to be mentioned for completeness is the OPC Unified Architecture (UA), which is an industrial machine-to-machine communication protocol [112]. It does not only provide mechanisms to transfer, but also semantically annotate data. However, until now, OPC UA is mainly used in industrial robotics but rarely in research.

One important point to notice is the communication overhead that results from a decentralized approach. Clearly, if software modules are distributed across different hardware modules, the time to transfer data between central controller and leaf modules must be considered. As discussed in Section 3.1.2, many different fieldbus or custom systems [103,113] are used in robotics. They can help to optimize the system regarding throughput, real-time requirements and QoS level on the physical and data link layers and are required to fulfill these requirements on the higher layers. Hence, the choice of appropriate communication hardware is essential to build distributed control systems.

4. Modeling and control

Multi-body kinematics and dynamics has been an area of extensive research during the past decades. While the term kinematics encompasses problems of position, velocity and acceleration analysis, the term dynamics refers to problems associated with the study of forces and torques and their effect on motion of multi-body systems. Kinematics and dynamics essentially forms the basis of behavior modeling and control of any robotic system. The usage of parallel submechanisms in a robot's design introduces a new level of complexity to their description, kinematics, dynamics and control. In this subsection, we discuss these domain specific difficulties and present some practical approaches used in controlling series-parallel hybrid robots.

4.1. Modeling

For describing serial robots, Denavit–Hartenberg (DH) parameters [114], and their modifications [115], have become the de-facto standard: they specify each coordinate transformation by only four parameters instead of six parameters, due to the particular placement of local coordinate systems at specific locations. In case of tree type robots and robots with closed loops, the traditional notion of DH parameters cannot be used and hence various extensions which may be used for describing hybrid robots have been proposed in the literature [116]. A comparison of various robot parameterization techniques can, for example, be found in Bongardt [117]. Due to the dependence of the frame placement on the link geometries, the modeling becomes unintuitive in particular for complex link shapes (for example in exoskeletons or human-machine interfaces). Another shortcoming of the extended DH notation for modeling hybrid robots is that it supports robots only with revolute or prismatic joint types. Higher DOF kinematic joints for example spherical joint should be decomposed into a set of three one DOF joints with virtual links in between in order to model them [118]. This is especially not practical for series-parallel hybrid robots with many parallel mechanisms containing passive spherical joints. Overall, finding extended DH parameters for a complex series-parallel hybrid robot is a tedious process which is error prone when performed manually.

For these reasons, standard open source robot description formats, such as `URDF`[3] (`ROS`), `COLLADA`[4] (`OpenRAVE`), or `SDF`[5] (`Gazebo`), do not rely on DH parameters (or extensions) for representing the coordinate transforms and, instead, store the required transformations by six parameters. These coordinate transforms requested by these description formats can be automatically extracted from CAD environments by programs such as `CAD-2-SIM` [119] and `SolidWorks to URDF Exporter`[6]. Even with the presence of such tools, it can be still time consuming to create complete robot description models for series-parallel hybrid robots because most formats do not allow the possibility of a modular description. Further, `URDF` does not allow for proper definition of closed loops, that often leads to complicated work-arounds when used for description of contemporary, complex robots. Other formats, such as `SDF`, allow the definition of parallel linkages, but do not further provide the functionality to explicitly define a spanning tree of a looped graph, necessary for a standardized tree representation of a model. Recently, the Lua based model description in the Rigid Body Dynamics Library [120] has been extended to allow the description of robots with closed loops while preserving the spanning tree representation of its topological graph. The Lua description in RBDL also allows the possibility to define higher DOF joints directly without the need of splitting them into simpler one DOF joint types. Moreover, within the D-RoCK [121] project which supported the development of Hybrid Robot Dynamics (`HyRoDyn`) software framework, the Supplementable, Mostly Universal Robot Format (`SMURF`[7]) was extended to allow a modular description of series-parallel hybrid robots [122].

4.2. Kinematics

Kinematics is often regarded as the study of *geometry of motion*. Direct or inverse geometric problems generally result in a set of non-linear algebraic equations regardless of the method of problem formulation. For serial robots, inverse geometric problem and for parallel robots, forward geometric problem are easy to formulate but difficult to solve. A rigorous kinematic analysis of generic series-parallel hybrid robots is an open problem because they carry kinematic complexity of both serial and parallel topologies. It is still quite appropriate to quote Nielsen and Roth [123] in this context: “Yet, a lot remains to be done before the subject of kinematic position analysis in robotics can be considered closed. Large structural classes, such as hybrid series and in-parallel systems, are just beginning to be treated in a systematic manner. Mainly, such studies are still done on a case-by-case basis, without a general theory and framework.” An example of such an analysis of 3-RPS-3-SPR type series-parallel hybrid manipulator can be found in Nayak et al. [124].

Nevertheless, a Jacobian based numerical and local solution to the geometric problems in singularity free regions is usually possible, which forms the basis of several multi-body simulation software. Also, developments in the field of modern kinematics [133] such as screw theory and computational algebraic geometry has helped researchers a great deal to study specific families of parallel or series-parallel hybrid mechanisms. Solutions to the inverse kinematics of a general 6R serial chain robot [134] and to the direct kinematics of the general Stewart-Gough platform [130], which yields polynomials of degree 16 and degree 40, respectively, are major advances due to application of computational algebraic geometry in this domain. Although such a comprehensive analysis requires some extra effort, the insights gained about the mechanism geometry, for example different assembly modes (solutions to forward kinematics problem), working modes (solutions to inverse kinematics problem), singularities in the workspace etc., outweighs the effort involved and hence these are preferred over numerical solutions for understanding their global behavior. Also, it is important to note that comprehensive kinematic analysis of most of the mechanisms listed in Tables 2 and 3 is readily available in the literature and has been summarized in Table 6. It must be noted that forward kinematics is usually solved with computational algebra tools which might not be feasible for real-time control but once it has been ensured that the mechanism's workspace is free of singularities, a Newton–Raphson based iterative approach can be used for solving the forward kinematics in real time.

4.3. Dynamics

Notable works in the field of robot dynamics include Newton–Euler [115,135], and Lagrangian [115] formulations, the Decoupled Natural Orthogonal Compliment (DeNOC) formulation [136], and Kane's method [137,138]. Traditionally, the equations of motion were described using 3D vectors – which quickly yields a large amount of equations for systems of connected bodies [139]. To address this issue, alternative compact and user-friendly formulations have been developed based on screw theory [135,140] and Lie group theory [141,142] which can easily be transformed into program code for modern computers.

Robots containing closed loops are subjected to additional geometric loop closure constraints which are difficult to resolve at position level and hence most multi-body dynamics software frameworks try to resolve them at velocity and acceleration levels and arrive at position constraints numerically. The Rigid Body Dynamics Library (`RBDL`) [143] and `OpenSim` [144] are some examples of open source libraries that implement such algorithms and import robot descriptions using various robot description formats. Another issue with series-parallel robots is the considerably large number of their spanning tree DOFs. Let n be the number of DOFs of the spanning tree representing a series-parallel hybrid robot, m be the total mobility of the robot, p be the number of actuators in the system and c be the number of independent closed loops. RH5 humanoid which only contains relatively simple parallel mechanism modules (with less than 3 DOF) has 32 DOF ($m = p = 32$), $c = 15$ independent closed loops and $n = 76$ DOF in its spanning tree. For a complicated hybrid robot such as Recupera-Reha exoskeleton, the robot has $m = p = 30$, $c = 29$ and $n = 128$. Hence, it can be challenging to solve the full dynamic model of such systems in real time. In most practical applications reported in literature, full dynamic model of the series-parallel hybrid robots is not exploited [41,65,145]. Recently, the development

[3] http://wiki.ros.org/urdf/XML.
[4] https://www.khronos.org/collada/.
[5] http://sdformat.org/.
[6] http://wiki.ros.org/sw_urdf_exporter.
[7] https://github.com/dfki-ric/phobos/wiki/smurf.

Table 6
Availability of kinematic analysis of some of the mechanisms presented in Tables 2 and 3 in the literature. Here, the methodology of the kinematic analysis is classified as algebraic or analytical and n denotes the maximum number of possible solutions.

Mechanism		Kinematic analysis	
Type	Mobility (m)	Forward method, n	Inverse method, n
1-$\underline{R}$RRR	1 DOF rotational	Analytical, 2 [125]	Analytical, 2 [125]
1-$\underline{R}$RPR	1 DOF rotational	Analytical, 2 [126]	Analytical, 1 [127]
2-S$\underline{P}$U+1U	2 DOF universal	Algebraic, 8 [66]	Analytical, 1 [20]
2-S$\underline{P}$RR+1U	2 DOF universal	Algebraic, 8 [66]	Analytical, 1 [66]
3-$\underline{R}$RR	3 DOF spherical	Analytical, 4 [128]	Analytical, 8 [128]
3-$\underline{R}$[2-SS]	3 DOF almost-spherical	Algebraic, 40 [129]	Analytical, 2 [69]
6-U$\underline{P}$S	6 DOF free	Algebraic, 40 [130]	Analytical, 1 [131]
6-$\underline{R}$US	6 DOF free	Algebraic, 40 [130]	Analytical, 32 [132]

of HyRoDyn has been reported [122] which is a modular software workbench for solving the kinematics and dynamics of series-parallel hybrid robots analytically. It does so by exploiting the closed form solutions to the loop constraints of typical parallel mechanisms used in the design of series-parallel hybrid robots like the ones presented in Table 6.

4.4. Control

More complex control strategies can significantly contribute to having better dynamic performance, i.e. closer performance to the expected theoretical one. However, historically the same classical control strategies used in serial robots have been reused for parallel or hybrid robots and there is relatively few specific literature on control of parallel devices [146] in comparison with their serial counterparts. The approaches followed basically fall into two categories: (a) the model-free control schemes such as PID control [147,148], fuzzy control [149], use neural networks to learn dynamics without a priori knowledge of the system [150] or based on force feedback such as in the seminal work in Merlet [151], and (b) model-based control schemes such optimal control [152] or the use of machine learning methods [153]. The use of same control methods as in serial robots leads to some limitations inherent to the nature of those methods. For instance, in serial robots it is a de facto standard to use joint space control (actuator space control to be precise), which is not the most suitable strategy for parallel robots and consequently, not for hybrid systems. A parallel robot is naturally described by its EE pose rather than by actuator configuration as in serial systems. For that reason, task space control is more suitable for parallel devices. Thus, for series-parallel hybrid robots, it is desirable to pick the generalized coordinates of the system as the set of EE coordinates of the parallel submechanism modules involved in the system instead of the set of actuator coordinates. This choice can lead to an analytical solution to the dynamic model of the overall system [122] as it is easier to solve the inverse kinematics of the parallel submechanism modules in comparison to their forward kinematics. However, some task space controllers may still require position and velocity information from the abstract joint space (refer to Fig. 8), which for parallel subsystems is much more complex to derive from the actuation space. This problem can be leveraged by utilizing parallel modules where the EE coordinates can be measured directly (refer Table 2) or a Newton–Raphson based iterative approach can be used for solving the forward kinematics in real time provided a singularity free workspace of the submechanism module.

Many series-parallel hybrid robots are kinematically controlled as they are relatively better suited for trajectory tracking applications due to superior accuracy of parallel submechanism modules. It is customary to implement the kinematic control of these systems in a decentralized way. For example, forward and inverse kinematics of the parallel submechanism modules can be solved on local controllers which communicate with the actuator control units (typically implementing PID controller) so that the overall system can be abstracted as a serial or tree-type system. The kinematics of the abstracted system is then solved on a central controller often combined with whole body control. Such decentralized kinematic control scheme is implemented in robots such as SherpaTT [154], MANTIS [23] and Charlie [22]. However, for applications such as exoskeletons or highly dynamic walking, it might become relevant to model the dynamics of the robot for purposes such as gravity compensation or computed torque control. This is usually done in a centralized manner (for e.g. Recupera exoskeleton [55]). To the best knowledge of the authors, a decentralized implementation of the *complete* dynamic model of the series-parallel hybrid robots, although theoretically feasible, has not been reported in the literature.

4.5. Adopted practices

Series-parallel hybrid robots are highly complex mechatronic systems and generic treatment of such robots remains an open problem. Hence, there is always a trade-off between modeling depth & accuracy and computational efficiency. However, modularity in robot design allows for certain abstractions which simplifies their modeling and control. Such abstractions are shown in Fig. 8. While Fig. 8(a) captures the true complexity of the robot, due to absence of generic methods to model and control such systems, three different abstractions are adopted to simplify the modeling and control. In the following, we discuss the practices used in design, modeling and control of series-parallel hybrid robots.

- Design: It is common practice to avoid any switch of assembly mode in the design of parallel submechanism modules for hybrid robots. It is achieved by choosing appropriate design parameters and physically restricting the movement of the joints in the parallel submechanism module. This ensures a unique forward kinematics solution for any given actuator input which makes the behavior of submechanism modules similar to serially connected joints and greatly simplifies the modeling and control of such systems. However, it comes at a cost of workspace restriction as certain kind of singularities can be crossed using appropriate trajectory planning in case of parallel robots [155].
- Kinematics: Forward and inverse kinematics of the submechanism module is usually solved to provide a bi-directional map between actuation space and abstract joint space (see Fig. 8(b) and (c)). Forward and inverse kinematics of parallel submechanism modules can be solved on local controllers either analytically or in resource-constrained systems with the help of Look Up Tables (LUTs). Analytical solutions are preferred when embedded hardware includes a microcontroller with a Floating Point Unit (FPU) (e.g. parallel joints in THOR [145]) or in cases when parallel submechanism modules bear more than two DOF (e.g. 6 DOF spine joint in Charlie [22]). As an alternative, LUTs can be used for systems without FPUs or FPGA-based local controllers (e.g. 1 or 2 DOF parallel joints in MANTIS [23] and 2 DOF ankle in Charlie [22]). Once a mapping is avail-

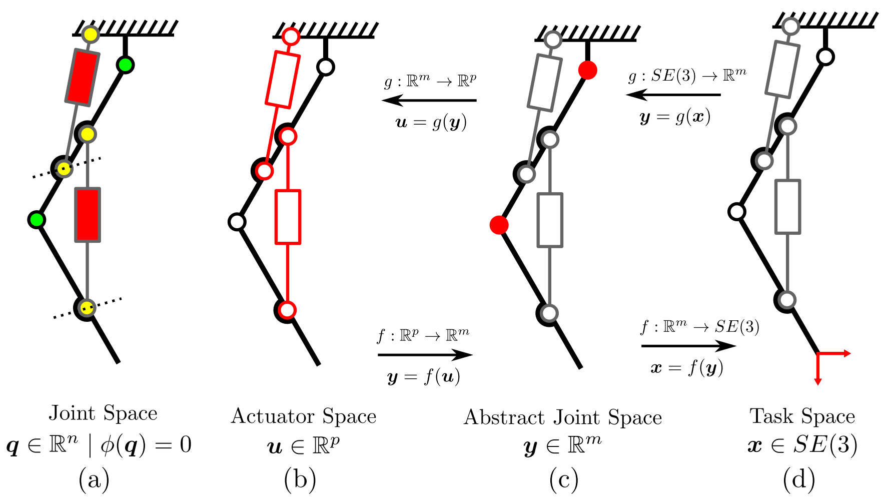


Fig. 8. Different abstractions used for modeling and control of hybrid robots.

able, the robot can be treated purely as a serial or tree type structure for which forward and inverse kinematics problems are easy to solve on the main controller(see Fig. 8(d)). Many series-parallel hybrid robots such as SherpaTT, MANTIS and Charlie are kinematically controlled and compliance is realized only with the help of force/torque measurements. Further, it is not common practice to compute the full kinematic state of the spanning tree (see Fig. 8(a)) since such calculations can be computationally expensive.

- Dynamics: As pointed out before, the computation of full inverse dynamic model for hybrid robots can be computationally very expensive due to the large size of their spanning trees and the large number of loop closure constraints to be resolved. The moving parts inside a parallel submechanism module may have relatively small contribution to the overall dynamics of the system which is essentially due to dynamics of link segments, and joint friction etc. [156]. Hence, an inverse dynamic model in abstract joint space is often combined with an inverse static model in actuation space to compute the actuator forces Jud et al. [50], Hopkins et al. [145], Vonwirth [157]. This approach is used in torque controlled series-parallel hybrid humanoids such as THOR, Valkyrie, Lola etc. The trade-off between the complete dynamic model and simplified dynamic model for series-parallel hybrid robots has been reported in Kumar et al. [158].

Overall, one could conclude that by adopting certain practices in design, modeling and control, it is possible to use such designs in various robotic applications and exploit the benefits offered by these architectures.

5. Discussion

Observing the development in robotics, it can be noticed that robots are becoming increasingly complex. To tackle this complexity, the design approach in robotics is moving from monolithic to modular. A modular robotic system is characterized by a finite number of scalable and reusable modules providing well-defined interfaces. Hence, this modularity must be reflected in the design of hardware (mechanical structure, electronics) and software components (low level controllers, kinematics and dynamic modeling, behavior modeling etc.) which constitute a robotic system. We argue that series-parallel hybrid robots have the potential to overcome the shortcomings of the traditional serial or tree type robots by providing a lightweight and compact construction, good dynamic characteristics and the inherent possibility to abstract higher DOF kinematic joints which leads to high modularity in robot architecture. We present a comparative study on humanoid leg design to support our argument.

Table 7
TORO ankle joint specification (length = 0.519 m, total weight of lower leg = 7.648 kg, weight of one actuator = 1.447 kg).

	Position	Max. abs. torque	Max. abs. velocity
Ankle pitch	−45° to 45°	130 N m	176° s^{-1}
Ankle roll	−19.5° to 19.5°	40 N m	120° s^{-1}

5.1. Comparative study

For the design of a humanoid, the designer typically tries to approximate the human anatomy with well studied kinematic joints. Anthropomorphic limb designs based on 6 DOF Universal-Revolute-Spherical (URS) or redundant 7 DOF Spherical-Revolute-Spherical (SRS) architecture are typically used in robotics. To demonstrate that a series-parallel hybrid design is better than a serial design, the lower leg design of RH5 humanoid is compared with TORO humanoid. Both of these humanoids are developed using a modular joint concept.

TORO's leg design mostly involves a serial arrangement of Light Weight Robot (LWR) drive units except that it uses a simple parallelogram mechanism for ankle pitch joint to reduce the overall leg inertia [36]. The pitch actuator (ILM85 weighing 2.062 kg) is placed closer to the knee joint with the help of a parallel four bar mechanism. The roll actuator (ILM50 weighing 0.832 kg) is then connected in series with the pitch guiding mechanism. The velocity and torque specification of the TORO ankle is listed in Table 7. The total weight of lower leg of TORO humanoid is 7.648 kg including the two actuators (total weight = 2.894 kg), foot unit, all the sensors, and electronics. Its total length measured from the knee axis to the foot sole is 0.519 m.

TORO's computing architecture corresponds to d) *central host with joint modules* category (see Table 4). Each drive unit contains sensors to measure joint torque and position and are controlled by a local controller [36]. All drive units are connected to one of TORO's central computers to receive desired positions and torques. TORO contains two central computers with different tasks (real-time control and high level planning/communication with drives, respectively). Furthermore, it contains two additional computers for stereo vision and ego motion/depth image computation. A Sercos-II ring bus with glass fibers is used for the communication between drive units and central PC. The kinematics and dynamics of the TORO humanoid is solved on the main CPU and the actuator commands are sent to the local actuator level controller.

TORO uses a proprietary middleware for control, called *Links and Nodes* [159,160]. The actual control software is generated by the Mathworks Realtime Workshop/Simulink Coder^{TM}. Regarding distributivity,

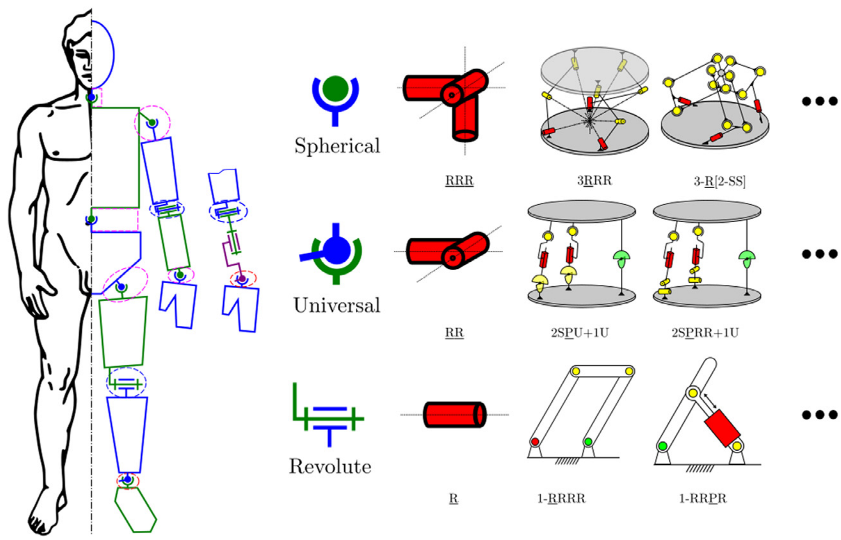


Fig. 9. Biologically inspired series-parallel hybrid humanoid design.

Table 8
RH5 Ankle v2.0 joint specification ($length$ = 0.525 m, total weight of lower leg = 3.6 kg, weight of one actuator = 0.610 kg).

Range (min. to max.)	Position	Max. abs. torque	Max. abs. velocity
Ankle pitch	−51.5° to 45°	120.9 N m to 304.2 N m	199.6° s^{-1} to 502.3° s^{-1}
Ankle roll	−57° to 57°	84.45 N m to 157.5 N m	385.9 ° s^{-1} to 726.1° s^{-1}
Linear actuator	221 mm to 331 mm	2000 N	265 mm s^{-1}

TORO distributes the overall functions (vision, planning, control) across the contained onboard computers. However, the real-time control as a single functions itself is not distributed across different computers.

On the other hand, RH5 humanoid [42] uses a series-parallel hybrid design involving the use of slider-crank linkage (1-RRPR) for hip flexion-extension joint & knee joint and a 2-SPRR + 1U orientational parallel mechanism for the ankle joint (refer Fig. 9). The idea is to strengthen certain joints for performing dynamic movements using parallel joint concept. It is well known that during human walking, the moments required for the ankle pitch movement are greater than ankle roll movements. Hence, instead of using a serial design, its better to use a parallel submechanism module encapsulating a universal joint. In 2-SPRR + 1U mechanism, when the two motors are actuated in the same direction, the mechanism produces a pitch only movement demonstrating good torque transmission characteristics. Further, it has been shown in biomechanics studies that during the ankle pitch movement of human gait, a peak torque between 105 Nm and 120 Nm is required when flexion/extension angle is between −6° and −12° [53]. The design parameters of the ankle mechanism were chosen such that maximum torque transmission occurs in this motion range. Table 8 presents the ankle joint specification for the RH5 humanoid based on the kinematic analysis presented in Kumar et al. [66]. The total weight of lower leg of RH5 humanoid is 3.6 kg including the foot unit, both actuators (total weight = 1.22 kg), all the sensors, and electronics. Its total length measured from the knee axis to the foot sole is 0.525 m.

The electronics architecture of RH5 belongs to the *e) combinational* category (see Fig. 4). For high level tasks, it contains a single main PC, which is connected to the joint modules via NDLCom [103], i.e., the communication architecture corresponds to a tree-like structure. Furthermore, it contains two NVidia®Jetson™boards to provide GPU-accelerated computations, e.g., for image processing and Deep Learning. The high-level and Jetson boards are connected via Ethernet. To communicate with the joint modules, an additional custom board containing a Xilinx®Zynq®(which his described in Wöhrle et al. [161]) implements the bridge between the high-level Ethernet-based communication and NDLCom. Each joint module is controlled by a Xilinx Spartan®6 FPGA-based cascaded local controller, which can provide a cascaded position, velocity and torque control. The parallel submechanisms in RH5 utilize a decentralized inverse kinematics using a look-up-table approach and dummyTXdummy– are implemented on the FPGAs of the joint modules. Hence, the system can be treated as a purely tree-type system for the purpose of kinematic control implemented on the main CPU. However, when the system is used in the force control mode, the full dynamic model of the system is solved on the main CPU and the joint modules receive the commanded forces/torques for a local actuator level control loop based on current/force measurements.

RH5 uses RoCK [102] as middleware and high-level control software. HyRoDyn [122] is used for kinematics and dynamics. Hence, due to RoCKs component based architecture it is in general possible to distribute the high-level functions across different processes and also computing components. However, RH5 currently uses the single main computer for all high-level control tasks. Nevertheless, kinematic operations are computed in a distributed fashion on heterogeneous devices, i.e., the kinematics for the parallel submechanism are computed on the local FPGAs.

Comparing Tables 7 and 8, it is clear that RH5 ankle can deliver better maximum velocity and torque for roll and pitch movements than TORO ankle without compromising the range of motion. Further, the weight of RH5 lower leg is almost half the weight of TORO lower leg while length of the two lower legs is almost equal. Thus, the use of a 2 DOF orientational parallel mechanism in the humanoid lower leg design is clearly advantageous over a serial design. A human-like range of motion can be easily achieved with good torque-velocity characteristics while minimizing the overall weight and leg inertia. Regarding the electronics architecture, both TORO as well as RH5 have several similarities. For example, they both rely on local controllers on the actuator level combined with one or more high level control computers. Since RH5 humanoid has several parallel joint modules, the kinematic control of the robot in the abstract joint space can be achieved in a fully decentralized way. However, the dynamic model of the robot is still solved on the central computer which can also potentially be distributed on the local controllers.

5.2. Modular and decentralized design paradigm

The above comparative study exemplifies the advantages of series-parallel hybrid design over a serial/tree-type design. It not only provides good dynamic characteristics and light weight design but also the possibility to abstract higher DOF kinematic joints with a modular design concept. Instead of using a commercial motor or actuator, it is now possible to use an off-the-shelf single DOF robotic joint with actuator unit, sensors, fully integrated power and control electronics (for e.g. ANYdrive [73], DFKI-X joint [92] etc.) which allows for a cascaded position, velocity and torque control directly in the joint space. In near future, it may also be possible to buy higher DOF joints (for e.g. universal, spherical etc., see Fig. 9) with integrated electronics and well defined control interfaces. Each joint can be unit-tested for its performance before integration and the system can be incrementally tested during the integration process. This kind of modular and decentralized design paradigm simplifies the robot development process and reduce the manufacturing costs and integration time. An excellent example of this design paradigm is the development of NASA's Valkyrie robot which was conceived, designed, manufactured, assembled, and verified in less than 12 months [35].

5.3. Challenges and future research directions

Building a robot is a sophisticated process which requires knowledge from different domains like mechanics, electronics, computer science etc. Hence, there is a strong need to bring together domain specific knowledge from the experts and made available to public use. There are many challenges to overcome for fully exploiting the potential of series-parallel hybrid designs. These challenges may be classified into theoretical and practical categories.

From a theoretical point of view, there is a need to develop methods for rigorous kinematic analysis of generic series-parallel hybrid mechanisms so that the properties of such systems can be better understood and exploited in design optimization. For example, techniques for solving their forward kinematics and inverse kinematics problems in a *global* sense and in *real-time* so that all possible solutions to these problems can be studied. Further, a general framework for studying their singularities in an exhaustive manner would be useful in understanding their behavior properly. The optimal synthesis problem for series-parallel hybrid robots is equally interesting. Given a high-level task description and a parameterized robot design, this problem concerns with finding the optimal morphological design parameters for the robot. An interesting recent work in this direction is [162] where it is shown how the design of a series-parallel hybrid robot can be optimized with simple linear actuators, four-bar linkages, or rotary servos. Extending such an approach to include general PKMs in the robot morphology would be an important milestone for the research community.

From a practical point of view, there is a strong need to develop a modular and distributable software framework for solving kinematics and dynamics of series-parallel hybrid robots so that complexity of submechanism modules catering to a specific motion subspace can be dealt with locally. A recent work in this direction is HyRoDyn software framework [122] which can solve kinematics and dynamics of series-parallel hybrid robots in a modular and analytical manner. Even though the algorithms involved are potentially distributable, it must be extended in order to work in a truly decentralized sense. One big hurdle one faces in this regard is the lack of a robotics middleware which works across platforms, heterogeneous processor types, and provides well defined data exchange interfaces. Most middlewares like ROS, RoCK etc still work on a standard CPU. Such a middleware would help in standardizing the design and implementation of distributable control techniques which are highly relevant for series-parallel hybrid robots. Lastly, there is also a need of software solution which will assist developers in the design of such robotic systems in a modular, composable and reusable fashion. RobMoSys [163] and D-RoCK [121]/Q-RoCK [121] are a few projects in this direction.

6. Conclusion

In the last two decades, an increasing number of series-parallel hybrid robot designs have emerged despite the challenges in their modeling and control. This paper presents a survey on series-parallel hybrid robots in different application domains such as humanoids, multi-legged robots, exoskeletons and industrial automation. Modular and distributive aspects of their design are discussed in both hardware and software domains. Theoretical challenges in modeling, kinematic and dynamic analysis, control of such robots are presented and approaches used in the literature to deal with these problems are discussed. Furthermore, with the help of a comparative study, we demonstrate the advantages of such designs and shed some light on the currently evolving modular and decentralized design paradigm in robotics. Challenges of dealing with such systems along with future research directions are identified.

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.

Acknowledgments

The work presented in this paper was performed within the projects D-RoCK and Q-RoCK, funded by the German Aerospace Center (DLR) with federal funds from the Federal Ministry of Education and Research (BMBF) (Grant numbers: 01IW15001 and FKZ 01IW18003 respectively). The fourth author acknowledges the support from the LCM K2 Center for Symbiotic Mechatronics within the framework of the Austrian COMET-K2 program.

References

[1] Robots U.. Collaborative Robots from Universal Robots. https://www.universal-robots.com/products/; Accessed 04.10.2019.

[2] Bischoff R, Kurth J, Schreiber G, Koeppe R, Albu-Schaeffer A, Beyer A, et al. The KUKA-DLR lightweight robot arm - a new reference platform for robotics research and manufacturing. In: ISR 2010 (41st International symposium on robotics) and ROBOTIK 2010 (6th German conference on robotics); 2010. p. 1–8.

[3] Robotics R.. Baxter. https://robots.ieee.org/robots/baxter/; Accessed 04.10.2019.

[4] ABB. YuMi - IRB 14,000—Collaborative Robot. https://new.abb.com/products/robotics/industrial-robots/irb-14000-yumi; Accessed 04.10.2019.

[5] Brinker J, Corves B. A survey on parallel robots with delta-like architecture. The 14th IFToMM World congress, 25.10.2015-30.10.2015, Taipei, Taiwan; 2015. doi:10.6567/IFToMM.14TH.WC.PS13.003.

[6] Stewart D. A platform with six degrees of freedom. Proc Inst Mech Eng 1965;180(1):371–86.

[7] Gosselin CM, Hamel J. The agile eye: a high-performance three-degree-of-freedom camera-orienting device. In: Proceedings of the 1994 IEEE International conference on robotics and automation, 1; 1994. p. 781–6. doi:10.1109/ROBOT.1994.351393.

[8] Patel Y, George P. Parallel manipulators applications—A survey. Mod Mech Eng 2012;2(03):57.

[9] Parallemic. Reviews in the Field of Parallel Mechanisms. https://www.parallemic.org/Reviews.html Accessed 04.10.2019.

[10] Luces M, Mills JK, Benhabib B. A review of redundant parallel kinematic mechanisms. J Intell Robot Syst 2017;86(2):175–98. doi:10.1007/s10846-016-0430-4.

[11] Sridhar BN. Methods for improving the knuckle boom capacity in truck-mounted cranes. J Manuf Sci Eng 1976;98(4):1183–7. doi:10.1115/1.3439079.

[12] Patel B, Prajapati J. A review on kinematics of hydraulic excavator's backhoe attachment. Int J Eng Sci Technol 2011;3:1990–7.

[13] Ipek L. Optimization of backhoe-loader mechanisms; 2006. Ph.D. thesis.

[14] Lee A, Horton David C, Willens KS. Wain-Roy and the invention of the backhoe. Leading Edge Attachments, Inc; 2018.

[15] Selig J. Geometric fundamentals of robotics. New York, NY: Springer New York; 2005. p. 1–9. doi:10.1007/0-387-27274-7_1.

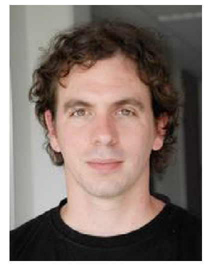
[16] To M, Webb P. An improved kinematic model for calibration of serial robots having closed-chain mechanisms. Robotica 2012;30(6):963–71. doi:10.1017/S0263574711001184.

[17] Gautier M, Khalil W, Restrepo PP. Identification of the dynamic parameters of a closed loop robot. In: Proceedings of 1995 IEEE International conference on robotics and automation, 3; 1995. p. 3045–50. doi:10.1109/ROBOT.1995.525717.
[18] Simnofske M, Kumar S, Bongardt B, Kirchner F. Active ankle - an almost-spherical parallel mechanism. 47th International symposium on robotics (ISR); 2016.
[19] Kong X, Gosselin CM. Type synthesis of parallel mechanisms. 1st ed. Springer Publishing Company, Incorporated; 2007.
[20] Serracn J, Puglisi L, Saltaren R, Ejarque G, Sabater-Navarro J, Aracil R. Kinematic analysis of a novel 2-d.o.f. orientation device. Robot Auton Syst 2012;60(6):852–61.
[21] Kumar S, Simnofske M, Bongardt B, Mueller A, Kirchner F. Integrating mimic joints into dynamics algorithms-exemplified by the hybrid Recupera exoskeleton. Advances in robotics (AIR-2017), June 28–July 2, New Delhi, India. ACM-ICPS; 2017.
[22] Kuehn D, Bernhard F, Burchardt A, Schilling M, Stark T, Zenzes M, et al. Distributed computation in a quadrupedal robotic system. Int J Adv Robot Syst 2014;11(7):110.
[23] Bartsch S, Manz M, Kampmann P, Dettmann A, Hanff H, Langosz M, et al. Development and control of the multi-legged robot mantis. In: Proceedings of ISR 2016: 47st International symposium on robotics; 2016. p. 1–8.
[24] Kirchner EA, Will N, Simnofske M, et al. Recupera-Reha: exoskeleton technology with integrated biosignal analysis for sensorimotor rehabilitation. In: Transdisziplinäre konferenz SmartASSIST; 2016. p. 504–17.
[25] Bai S, Christensen S, Islam MRU. An upper-body exoskeleton with a novel shoulder mechanism for assistive applications. In: 2017 IEEE International conference on advanced intelligent mechatronics (AIM); 2017. p. 1041–6. doi:10.1109/AIM.2017.8014156.
[26] Lohmeier S, Buschmann T, Ulbrich H, Pfeiffer F. Modular joint design for performance enhanced humanoid robot LOLA. In: Proceedings 2006 IEEE International conference on robotics and automation, 2006. ICRA 2006.; 2006. p. 88–93.
[27] Lee B., Knabe C., Orekhov V., Hong D. Design of a human-like range of motion hip joint for humanoid robots. Proceedings of the ASME 2014 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. Volume 5B: 38th Mechanisms and Robotics Conference. Buffalo, New York, USA. August 17-20, 2014. V05BT08A018. ASME. https://doi.org/10.1115/DETC2014-35214.
[28] Fukuda T, Nakagawa S. Dynamically reconfigurable robotic system. In: Proceedings 1988 IEEE International conference on robotics and automation, 3; 1988. p. 1581–6. doi:10.1109/ROBOT.1988.12291.
[29] Yim M, Shen Wm, Salemi B, Rus D, Moll M, Lipson H, et al. Modular self-reconfigurable robot systems [grand challenges of robotics]. IEEE Robot Autom Mag 2007;14(1):43–52. doi:10.1109/MRA.2007.339623.
[30] Ahmadzadeh H, Masehian E, Asadpour M. Modular robotic systems: characteristics and applications. J Intell Robot Syst 2016;81(3):317–57. doi:10.1007/s10846-015-0237-8.
[31] Chennareddy SSR, Agrawal A, Karuppiah A. Modular self-reconfigurable robotic systems: a survey on hardware architectures. J Robot 2017. doi:10.1155/2017/5013532.
[32] Stasse O, Flayols T. An overview of humanoid robots technologies. Cham: Springer International Publishing; 2019. p. 281–310. doi:10.1007/978-3-319-93870-7_13.
[33] Lohmeier S. Design and realization of a humanoid robot for fast and autonomous bipedal locomotion. Technischen Universität Mnchen; 2010. Ph.D. thesis.
[34] Lemburg J, de Gea Fernández J, Eich M, Mronga D, Kampmann P, Vogt A, et al. AILA - design of an autonomous mobile dual-arm robot. In: 2011 IEEE International conference on robotics and automation (ICRA); 2011. p. 5147–53. doi:10.1109/ICRA.2011.5979775.
[35] Radford NA, Strawser P, Hambuchen K, Mehling J, et al. Valkyrie: NASA's first bipedal humanoid robot. J Field Robot 2015;32(3):397–419. doi:10.1002/rob.21560.
[36] Englsberger J, Werner A, Ott C, Henze B, Roa MA, Garofalo G, et al. Overview of the torque-controlled humanoid robot TORO. In: 2014 IEEE-RAS International conference on humanoid robots; 2014. p. 916–23.
[37] Stasse O, Flayols T, Budhiraja R, Giraud-Esclasse K, Carpentier J, Mirabel J, et al. TALOS: a new humanoid research platform targeted for industrial applications. In: 2017 IEEE-RAS 17th International conference on humanoid robotics (Humanoids); 2017. p. 689–95. doi:10.1109/HUMANOIDS.2017.8246947.
[38] Lee BKT-S. Design of a humanoid robot for disaster response. Virginia Tech; 2014. Ph.D. thesis.
[39] Lahr DF, Yi H, Hong DW. Biologically inspired design of a parallel actuated humanoid robot. Adv Robot 2016;30(2):109–18. doi:10.1080/01691864.2015.1094408.
[40] Cafolla D, Wang M, Carbone G, Ceccarelli M. LARMBot: a new humanoid robot with parallel mechanisms. In: Parenti-Castelli V, Schiehlen W, editors. ROMANSY 21 - robot design, dynamics and control. Cham: Springer International Publishing; 2016. p. 275–83.
[41] Schütz S, Nejadfard A, Mianowski K, Vonwirth P, Berns K. CARL—a compliant robotic leg featuring mono- and biarticular actuation. In: IEEE-RAS International conference on humanoid robots; 2017.
[42] Peters H, Kampmann P, Simnofske M. Konstruktion eines zweibeinigen humanoiden roboters. 2. VDI Fachkonferenz humanoide roboter; 2017.
[43] Gim KG, Kim J, Yamane K. Design and fabrication of a bipedal robot using serial-parallel hybrid leg mechanism. In: 2018 IEEE/RSJ International conference on intelligent robots and systems (IROS); 2018. p. 5095–100. doi:10.1109/IROS.2018.8594182.
[44] Gim KG, Kim J, Yamane K. Design of a serial-parallel hybrid leg for a humanoid robot. In: 2018 IEEE International conference on robotics and automation (ICRA); 2018. p. 1–5. doi:10.1109/ICRA.2018.8460733.
[45] Cordes F, Oekermann C, Babu A, Kuehn D, Stark T, Kirchner F. An active suspension system for a planetary rover. In: Proceedings of the international symposium on artificial intelligence, robotics and automation in space (i-SAIRAS 2014) Montreal, Canada, o.A.; 2014.
[46] Cordes F, Babu A, Kirchner F. Static force distribution and orientation control for a rover with an actively articulated suspension system. In: Proceedings of the 2017 IEEE/RSJ International conference on intelligent robots and systems. IEEE/RSJ International conference on intelligent robots and systems (IROS-17), friendly people, friendly robots, September 24–28, Vancouver, BC, Canada. IEEE/RSJ; 2017.
[47] Townsend J, Biesiadecki J, Collins C. Athlete mobility performance with active terrain compliance. In: 2010 IEEE Aerospace conference; 2010. p. 1–7. doi:10.1109/AERO.2010.5446992.
[48] Ceccarelli M, Cafolla D, Russo M, Carbone G. Heritagebot platform for service in cultural heritage frames. International Journal of Advanced Robotic Systems 2018;15(4). doi:10.1177/1729881418790692. 1729881418790692
[49] Jud D, Hottiger G, Leemann P, Hutter M. Planning and control for autonomous excavation. IEEE Robot Autom Lett 2017;2(4):2151–8. doi:10.1109/LRA.2017.2721551.
[50] Jud D, Leemann P, Kerscher S, Hutter M. Autonomous free-form trenching using a walking excavator. IEEE Robot Autom Lett 2019;4(4):3208–15. doi:10.1109/LRA.2019.2925758.
[51] Wensing PM, Wang A, Seok S, Otten D, Lang J, Kim S. Proprioceptive actuator design in the mit cheetah: impact mitigation and high-bandwidth physical interaction for dynamic legged robots. IEEE Trans Robot 2017;33(3):509–22. doi:10.1109/TRO.2016.2640183.
[52] Dholakiya D, Bhattacharya S, Gunalan A, Singla A, Bhatnagar S, Amrutur B, et al. Design, development and experimental realization of a quadrupedal research platform: stoch. In: 2019 5th International conference on control, automation and robotics (ICCAR); 2019.
[53] Zoss AB, Kazerooni H, Chu A. Biomechanical design of the berkeley lower extremity exoskeleton (BLEEX). IEEE/ASME Trans Mechatron 2006;11(2):128–38.
[54] Bionics B.. ExoHiker. https://bleex.me.berkeley.edu/research/exoskeleton/exohiker/; Accessed: 10.09.2019.
[55] Kumar S, Wöhrle H, Trampler M, Simnofske M, Peters H, Mallwitz M, et al. Modular design and decentralized control of the Recupera exoskeleton for stroke rehabilitation. Appl Sci 2019;9(4):626.
[56] Kim B, Deshpande AD. An upper-body rehabilitation exoskeleton harmony with an anatomical shoulder mechanism: design, modeling, control, and performance evaluation. Int J Robot Res 2017;36(4):414–35. doi:10.1177/0278364917706743.
[57] Sadeqi S, Bourgeois SP, Park EJ, Arzanpour S. Design and performance analysis of a 3-RRR spherical parallel manipulator for hip exoskeleton applications. J Rehabil Assist Technol Eng 2017;4. doi:10.1177/2055668317697596. 2055668317697596
[58] Merlet JP. Structural synthesis and architectures. Dordrecht: Springer Netherlands; 2006. p. 19–94. doi:10.1007/1-4020-4133-0_2.
[59] Charentus S. Modélisation et commande d'un robot manipulateur redondant compose de plusierus plates-formes de stewart. UPS, Toulouse; 1990. Ph.D. thesis.
[60] Brinker J, Funk N, Ingenlath P, Takeda Y, Corves B. Comparative study of serial-parallel delta robots with full orientation capabilities. IEEE Robot Autom Lett 2017;2(2):920–6. doi:10.1109/LRA.2017.2654551.
[61] Hildebrandt M, Albiez J, Kirchner F. Computer-based control of deep-sea manipulators. In: OCEANS 2008 - MTS/IEEE Kobe techno-ocean; 2008. p. 1–6. doi:10.1109/OCEANSKOBE.2008.4531026.
[62] Hildebrandt M, Albiez J, Fritsche M, Hilljegerdes J, Kloss P, Wirtz M, et al. Design of an autonomous under-ice exploration system. In: 2013 OCEANS - San Diego; 2013. p. 1–6. doi:10.23919/OCEANS.2013.6741164.
[63] Hubicki C, Grimes J, Jones M, Renjewski D, Spr witz A, Abate A, et al. ATRIAS: design and validation of a tether-free 3D-capable spring-mass bipedal robot. Int J Robot Res 2016;35(12):1497–521. doi:10.1177/0278364916648388.
[64] Garcia E, Arevalo JC, Muoz G, Gonzalez-de Santos P. On the biomimetic design of agile-robot legs. Sensors 2011;11(12):11305–34. doi:10.3390/s111211305.
[65] Nejadfard A, Schütz S, Vonwirth P, Mianowski K, Karsten B. Moment arm analysis of the biarticular actuators in compliant robotic leg CARL. In: Conference on biomimetic and biohybrid systems. Springer: Springer International Publishing; 2018. p. 348–60.
[66] Kumar S, Nayak A, Peters H, Schulz C, Mueller A, Kirchner F. Kinematic analysis of a novel parallel 2SPRR + 1U ankle mechanism in humanoid robot. Advances in robot kinematics. Carricato M, editor. Cham: Springer Verlag GmbH; 2018.
[67] Parmiggiani A, Fiorio L, Scalzo A, Sureshbabu AV, Randazzo M, Maggiali M, et al. The design and validation of the R1 personal humanoid. In: 2017 IEEE/RSJ International conference on intelligent robots and systems (IROS); 2017. p. 674–80. doi:10.1109/IROS.2017.8202224.
[68] Kuo C-H, Dai JS. Task-oriented structure synthesis of a class of parallel manipulators using motion constraint generator. Mech Mach Theory 2013;70:394–406.
[69] Kumar S, Bongardt B, Simnofske M, Kirchner F. Design and kinematic analysis of the novel almost spherical parallel mechanism active ankle. J Intell Robot Syst 2019;94:303–25. doi:10.1007/s10846-018-0792-x.
[70] Kuehn D, Dettmann A, Kirchner F. Analysis of using an active artificial spine in a quadruped robot. In: 2018 4th International conference on control, automation and robotics (ICCAR). International conference on control, automation and robotics (ICCAR-2018), April 20–23, Auckland, New Zealand. IEEE Xplore online; 2018.
[71] Fiorio L, Scalzo A, Natale L, Metta G, Parmiggiani A. A parallel kinematic mechanism for the torso of a humanoid robot: design, construction and validation. In: 2017 IEEE/RSJ International conference on intelligent robots and systems (IROS); 2017. p. 681–8. doi:10.1109/IROS.2017.8202225.
[72] Sureshbabu AV, Chang JH, Fiorio L, Scalzo A, Metta G, Parmiggiani A. A parallel kinematic wrist for the R1 humanoid robot. In: 2017 IEEE Interna-

tional conference on advanced intelligent mechatronics (AIM); 2017. p. 1215–20. doi:10.1109/AIM.2017.8014184.
[73] ANYbotics. ANYdrive. https://www.anybotics.com/anydrive/ Accessed: 15.05.2018.
[74] Robotis. Dynamixel series. http://www.robotis.us/dynamixelAccessed: 22.08.2018.
[75] Robotis. Dynamixel pro series. http://www.robotis.us/dynamixel-pro; Accessed: 22.08.2018.
[76] Kinova. Kinovatmactuator. https://www.kinovarobotics.com/en/products/actuator-series.
[77] TREE. Technology for Robotic systems Entering real Environments (TREE) Actuators. http://www.treerobotics.eu/tree-robotics-actuators-info-sheet/file; Accessed: 10.09.2018.
[78] Rader S, Kaul L, Weiner P, Asfour T. Highly integrated sensor-actuator-controller units for modular robot design. In: 2017 IEEE International conference on advanced intelligent mechatronics (AIM); 2017. p. 1160–6. doi:10.1109/AIM.2017.8014175.
[79] Bridgwater LB, Ihrke C, Diftler MA, Abdallah ME, Radford NA, Rogers J, et al. The Robonaut 2 hand-designed to do work with tools. In: International conference on robotics and automation (ICRA). IEEE; 2012. p. 3425–30.
[80] Bodie K, Bellicoso CD, Hutter M. ANYpulator: design and control of a safe robotic arm. In: 2016 IEEE/RSJ International conference on intelligent robots and systems (IROS). IEEE; 2016. p. 1119–25.
[81] Hutter M, Gehring C, Jud D, Lauber A, Bellicoso CD, Tsounis V, et al. ANYmal-a highly mobile and dynamic quadrupedal robot. In: 2016 IEEE/RSJ International conference on intelligent robots and systems (IROS). IEEE; 2016. p. 38–44.
[82] Jäntsch M, Wittmeier S, Knoll A. Distributed control for an anthropomimetic robot. In: 2010 IEEE/RSJ International conference on intelligent robots and systems; 2010. p. 5466–71. doi:10.1109/IROS.2010.5651169.
[83] PROFIBUS & PROFINET International (PI). PROFIBUS. https://www.profibus.com/technology/profibus/; Accessed: 6.9.2018.
[84] Gutiérrez C.S.V., Juan L.U.S., Ugarte I.Z., Vilches V.M.. Time-sensitive networking for robotics. CoRR abs/1804.076432018.
[85] PROFIBUS & PROFINET International (PI). PROFINET the leading industrial ethernet standard. https://www.profibus.com/technology/profinet/; Accessed: 6.9.2018.
[86] EtherCAT Technology Group. EtherCAT - the ethernet fieldbus. https://www.ethercat.org/en/technology.html; Accessed: 2018-09-06.
[87] Sercos International e. V. SERCOS - the automation bus. https://www.sercos.orgAccessed: 2018-09-06.
[88] Albu-Schäffer A, Haddadin S, Ott C, Stemmer A, Wimböck T, Hirzinger G. The DLR lightweight robot: design and control concepts for robots in human environments. Ind Robot 2007;34(5):376–85. doi:10.1108/01439910710774386.
[89] Kuehn D, Schilling M, Stark T, Zenzes M, Kirchner F. System design and field testing of the hominid robot charlie. J Field Robot 2016;34(Issue 4):666–703.
[90] Negrello F, Garabini M, Catalano MG, Malzahn J, Caldwell DG, Bicchi A, et al. A modular compliant actuator for emerging high performance and fall-resilient humanoids. In: 2015 IEEE-RAS 15th International conference on humanoid robots (Humanoids); 2015. p. 414–20. doi:10.1109/HUMANOIDS.2015.7363567.
[91] Hebert P, Bajracharya M, Ma J, Hudson N, Aydemir A, Reid J, et al. Mobile manipulation and mobility as manipulation-design and algorithms of robosimian. J Field Robot 2015;32(2):255–74. doi:10.1002/rob.21566.
[92] Sonsalla R, Hanff H, Schöberl P, Stark T, Mulsow NA. DFKI-X: a novel, compact and highly integrated robotics joint for space applications. In: Proceedings of the 17th European space mechanisms and tribology symposium. European space mechanisms and tibology symposium (ESMATS-2017), 17th, September 20–22, Hatfield, United Kingdom. ESMATS; 2017.
[93] Ambrose RO, Aldridge H, Askew RS, Burridge RR, Bluethmann W, Diftler M, et al. Robonaut: NASA's space humanoid. IEEE Intell Syst Appl 2000;15(4):57–63.
[94] Diftler MA, Mehling J, Abdallah ME, Radford NA, Bridgwater LB, Sanders AM, et al. Robonaut 2-the first humanoid robot in space. In: Robotics and automation (ICRA), 2011 IEEE International conference on. IEEE; 2011. p. 2178–83.
[95] Mohamed N, Al-Jaroodi J, Jawhar I. A review of middleware for networked robots. Int J Comput Sci Netw Secur 2009;9(5):139–48.
[96] Elkady A, Sobh T. Robotics middleware: a comprehensive literature survey and attribute-based bibliography. J Robot 2012;2012:1–15. doi:10.1155/2012/959013.
[97] Quigley M, Conley K, Gerkey BP, Faust J, Foote T, Leibs J, et al. ROS: an open–source robot operating system ICRA Workshop on open source software; 2009.
[98] Metta G, Fitzpatrick P, Natale L. YARP: yet another robot platform. Int J Adv Robot Syst 2006;3(1):8. doi:10.5772/5761.
[99] Aragão M, Moreno P, Bernardino A. Middleware interoperability for robotics: aROS YARP framework. Front Robot AI 2016;3:64. doi:10.3389/frobt.2016.00064.
[100] Bruyninckx H. Open robot control software: the OROCOS project. In: Proceedings 2001 ICRA. IEEE International conference on robotics and automation (Cat. No.01CH37164), 3; 2001. p. 2523–8. doi:10.1109/ROBOT.2001.933002.
[101] Bruyninckx H, Soetens P, Koninckx B. The real-time motion control core of the OROCOS project. In: 2003 IEEE International conference on robotics and automation (Cat. No.03CH37422), 2; 2003. p. 2766–71. doi:10.1109/ROBOT.2003.1242011.
[102] Joyeux S.. Rock: the robot construction kit. https://www.rock-robotics.org/stable/; Accessed: 10.05.2018.
[103] Zenzes M, Kampmann P, Stark T, Schilling M. NdlCom: simple protocol for heterogeneous embedded communication networks. In: Proceedings of the embedded world exhibition & conference, Nuremberg, Germany; 2016. p. 23–5.
[104] Reichardt M, Föhst T, Berns K. Introducing FINROC: a convenient real-time framework for robotics based on a systematic design approach. Tech. Rep.. Robotics Research Lab, Department of Computer Science, University of Kaiserslautern, Kaiserslautern, Germany; 2012.
[105] Reichardt M, Schütz S, Berns K. One fits more–on highly modular quality-driven design of robotic frameworks and middleware. J Softw Eng Robot 2017;8(1):141–53.
[106] Open Source Robotics Foundation (OSRF). ROS2. https://github.com/ros2/
[107] Google. Protocol Buffers. https://developers.google.com/protocol-buffers/; Accessed 18.5.2018.
[108] iMatix. ZeroMQ distributed messaging. http://zeromq.org/; Accessed: 18.5.2018.
[109] Pardo-Castellote G. OMG data-distribution service: architectural overview. In: 23rd International conference on distributed computing systems workshops, 2003. Proceedings; 2003. p. 200–6. doi:10.1109/ICDCSW.2003.1203555.
[110] Schlesselman JM, Pardo-Castellote G, Farabaugh B. OMG data-distribution service (DDS): architectural update. In: Military communications conference, 2. IEEE; 2004. p. 961–7.
[111] Maruyama Y, Kato S, Azumi T. Exploring the performance of ROS2. In: 2016 International conference on embedded software (EMSOFT); 2016. p. 1–10. doi:10.1145/2968478.2968502.
[112] Mahnke W, Leitner S-H, Damm M. OPC unified architecture. Springer Science & Business Media; 2009.
[113] Zhang L, Slaets P, Bruyninckx H. An open embedded hardware and software architecture applied to industrial robot control. In: 2012 IEEE International conference on mechatronics and automation; 2012. p. 1822–8. doi:10.1109/ICMA.2012.6285098.
[114] Hartenberg D, Scheunemann R. A kinematic notation for lower-pair mechanisms based on matrices. Trans ASME J Appl Mech 1955;22:215–21.
[115] Khalil W, Dombre E. Modeling, identification and control of robots. 3rd ed. Bristol, PA, USA: Taylor & Francis, Inc.; 2002.
[116] Khalil W, Kleinfinger J. A new geometric notation for open and closed-loop robots. In: Proceedings. 1986 IEEE International conference on robotics and automation, 3; 1986. p. 1174–9. doi:10.1109/ROBOT.1986.1087552.
[117] Bongardt B. Sheth uicker convention revisited. Mech Mach Theory 2013;69:200–29.
[118] Shah SV, Saha SK, Dutt JK. Denavit-Hartenberg parameterization of euler angles. J Comput Nonlinear Dyn 2012;7(2):21006. doi:10.1115/1.4005467.
[119] Bongardt B. CAD-2-SIM – kinematic modeling of mechanisms based on the sheth-uicker convention. Berlin, Heidelberg: Springer Berlin Heidelberg; 2011. p. 465–77.
[120] Martin Felis. RBDL Lua Models with Constraints. https://rbdl.bitbucket.io/d8/d35/addon_luamodel_page.html#luamodel_introduction; Accessed: 6.9.2019.
[121] D-RoCK. Models, methods and tools for the model based software development of robots. https://robotik.dfki-bremen.de/en/research/projects/d-rock.html; Accessed: 10.05.2018.
[122] Kumar S, Szadkowski KAV, Mueller A, Kirchner F. An Analytical and Modular Software Workbench for Solving Kinematics and Dynamics of Series-Parallel Hybrid Robots. ASME. J. Mechanisms Robotics. 2020;12(2):021114. https://doi.org/10.1115/1.4045941
[123] Nielsen J, Roth B. On the kinematic analysis of robotic mechanisms. Int J Robot Res 1999;18(12):1147–60. doi:10.1177/02783649922067771.
[124] Nayak A, Caro S, Wenger P. Kinematic analysis of the 3-RPS-3-SPR series–parallel manipulator. Robotica 2019;37(7):1240–66. doi:10.1017/S0263574718000826.
[125] Bai S, Angeles J. A unified input-output analysis of four-bar linkages. Mechanism and Machine Theory 2008;43(2):240–51. doi:10.1016/j.mechmachtheory.2007.01.002. http://www.sciencedirect.com/science/article/pii/S0094114X07000249
[126] Featherstone R. Closed loop systems. Boston, MA: Springer US; 2008. p. 141–69. doi:10.1007/978-1-4899-7560-7_8.
[127] Kumar S. Modular and Analytical Methods for Solving Kinematics and Dynamics of Series-Parallel Hybrid Robots. Universität Bremen; 2019. Ph.D. thesis.
[128] Bonev IA, Chablat D, Wenger P. Working and assembly modes of the agile eye. In: Proceedings 2006 IEEE International conference on robotics and automation, 2006. ICRA 2006.; 2006. p. 2317–22. doi:10.1109/ROBOT.2006.1642048.
[129] Kumar S, Nayak A, Bongardt B, Mueller A, Kirchner F. Kinematic analysis of active ankle using computational algebraic geometry. In: Zeghloul S, Romdhane L, Laribi MA, editors. Computational kinematics. Cham: Springer International Publishing; 2018. p. 117–25.
[130] Husty M. An algorithm for solving the direct kinematics of general Stewart-Gough platforms. Mech Mach Theory 1996;31(4):365–79. doi:10.1016/0094-114X(95)00091-C.
[131] Lynch KM, Park FC. Modern robotics: mechanics, planning, and control. 1st ed. New York, NY, USA: Cambridge University Press; 2017.
[132] Dutta A, Salunkhe DH, Kumar S, Udai AD, Shah SV. Sensorless full body active compliance in a 6 DOF parallel manipulator. Robotics and Computer-Integrated Manufacturing 2019;59:278–90. doi:10.1016/j.rcim.2019.04.010. http://www.sciencedirect.com/science/article/pii/S0736584518303296
[133] McCarthy JM. Polynomials, computers, and kinematics for the 21st century. In: McCarthy JM, editor. 21st Century kinematics. London: Springer London; 2013. p. 1–12.
[134] Raghavan M, Roth B. Inverse kinematics of the general 6R manipulator and related linkages. J Mech Des 1993;115(3):502–8.
[135] Roy Featherstone. *Rigid Body Dynamics Algorithms*. Berlin, Heidelberg: Springer-Verlag; 2007.
[136] Rao AK, Saha S, Rao P. Dynamics modelling of hexaslides using the decoupled natural orthogonal complement matrices. Multibody Syst Dyn 2006;15(2):159–80.
[137] Lesser M. A geometrical interpretation of Kane's equations. Proc Math Phys Sci 1992;436(1896):69–87.
[138] Buffinton KW. Robotics and automation handbook. CRC Press; 2005.

[139] Luh JYS, Walker MW, Paul RP. On-line computational scheme for mechanical manipulators. ASME J Dyn Syst Meas Control 1980;102(2):69–76. doi:10.1115/1.3149599.
[140] Jain A. Robot and multibody dynamics: analysis and algorithms. Springer Verlag; 2011.
[141] Müller A. Screw and lie group theory in multibody kinematics. Multibody Syst Dyn 2017. doi:10.1007/s11044-017-9582-7.
[142] Müller A. Screw and lie group theory in multibody dynamics. Multibody Syst Dyn 2018;42(2):219–48. doi:10.1007/s11044-017-9583-6.
[143] Felis ML. RBDL: an efficient rigid-body dynamics library using recursive algorithms. Auton Robots 2017;41(2):495–511.
[144] Delp SL, Anderson FC, Arnold AS, Loan P, Habib A, John CT, et al. Opensim: open-source software to create and analyze dynamic simulations of movement. IEEE Trans Biomed Eng 2007;54(11):1940–50. doi:10.1109/TBME.2007.901024.
[145] Hopkins MA, Ressler SA, Lahr DF, Leonessa A, Hong DW. Embedded joint-space control of a series elastic humanoid. In: 2015 IEEE/RSJ International conference on intelligent robots and systems (IROS); 2015. p. 3358–65. doi:10.1109/IROS.2015.7353845.
[146] Ghorbel FH, Chetelat O, Gunawardana R, Longchamp R. Modeling and set point control of closed-chain mechanisms: theory and experiment. IEEE Trans Control Syst Technol 2000;8(5):801–15. doi:10.1109/87.865853.
[147] Khosravi MA, Taghirad HD. Robust PID control of fully-constrained cable driven parallel robots. Mechatronics 2014;24(2):87–97. doi:10.1016/j.mechatronics.2013.12.001.
[148] Su Y, Duan B, Zheng C. Nonlinear PID control of a six-DOF parallel manipulator. IEE Proc - Control Theory Appl 2004;151. 95–102(7)
[149] Zi B, Duan B, Du J, Bao H. Dynamic modeling and active control of a cable-suspended parallel robot. Mechatronics 2008;18(1):1–12. doi:10.1016/j.mechatronics.2007.09.004.
[150] Mirza MA, Li S, Jin L. Simultaneous learning and control of parallel stewart platforms with unknown parameters. Neurocomputing 2017;266:114–22. doi:10.1016/j.neucom.2017.05.026.
[151] Merlet JP. Force-feedback control of parallel manipulators. In: Proceedings 1988 IEEE International conference on robotics and automation, 3; 1988. p. 1484–9. doi:10.1109/ROBOT.1988.12277.
[152] Liu GF, Cheng H, Xiong ZH, Wu XZ, Wu YL, Li ZX. Distribution of singularity and optimal control of redundant parallel manipulators. In: Proceedings 2001 IEEE/RSJ International conference on intelligent robots and systems. Expanding the societal role of robotics in the next millennium (Cat. No. 01CH37180), 1; 2001. p. 177–82. doi:10.1109/IROS.2001.973355.
[153] Abdellatif H, Heimann B. Advanced model-based control of a 6-DOF hexapod robot: a case study. IEEE/ASME Trans Mechatron 2010;15(2):269–79. doi:10.1109/TMECH.2009.2024682.
[154] Cordes F, Kirchner F, Babu A. Design and field testing of a rover with an actively articulated suspension system in a mars analog terrain. J Field Robot 2018;35(7):1149–81. doi:10.1002/rob.21808.
[155] Pagis G, Bouton N, Briot S, Martinet P. Enlarging parallel robot workspace through type-2 singularity crossing. Control Eng Pract 2015;39:1–11. doi:10.1016/j.conengprac.2015.01.009.
[156] Buschmann T, Favot V, Schwienbacher M, Ewald A, Ulbrich H. Dynamics and control of the biped robot LOLA. In: Gattringer H, Gerstmayr J, editors. Multibody system dynamics, robotics and control. Vienna: Springer Vienna; 2013. p. 161–73.
[157] Vonwirth P. Modular control architecture for bipedal walking on a single compliant leg. Robotics Research Lab, University of Kaiserslautern; 2017. Master's thesis. Unpublished; supervised by Steffen Schütz
[158] Kumar S, Martensen J, Mueller A, Kirchner F. Model simplification in dynamic control of series-parallel hybrid robots—A representative study of the neglected dynamics. In: 2019 IEEE/RSJ International conference on intelligent robots and systems (IROS); 2019.
[159] Burger Schmidt F, Burger R. How we deal with software complexity in robotics: "links and nodes" and the "robotkernel". Tech. Rep.. DLR - Institute of Robotics and Mechatronics; 2014.
[160] Burger R, Schmidt F, Brunner B. Development of a runtime-configurable robotic hardware abstraction framework. Tech. Rep.. Astra 2015 RCOS Forum; 2015.
[161] Wöhrle H, Tabie M, Kim SK, Kirchner E, Kirchner F. A hybrid FPGA-based system for EEG- and EMG-based online movement prediction. Sensors 2017(17). doi:10.3390/s17071552.
[162] Ha S, Coros S, Alspach A, Kim J, Yamane K. Computational co-optimization of design parameters and motion trajectories for robotic systems. Int J Robot Res 2018;37(13–14):1521–36. doi:10.1177/0278364918771172.
[163] RobMoSys. RobMoSys: Composable Models and Software for Robotic Systems. https://robmosys.eu/; Accessed: 10.05.2018.

Shivesh Kumar is a researcher and team leader of the Mechanics & Control team at the Robotics Innovation Center, German Research Center for Artificial Intelligence (DFKI GmbH), Bremen. He obtained his PhD degree (Dr. rer. nat.) from Faculty of Mathematics and Computer Science at the University of Bremen in November 2019. Previously, he holds a Masters degree in Control Engineering, Robotics, and Applied Informatics with specialization in Advanced Robotics from Ecole Centrale de Nantes, France in 2015 and Bachelors degree in Mechanical Engineering from National Institute of Technology Karnataka, India in 2013. His research interests include kinematics, dynamics and control of robots with applications in the fields of exoskeletons, humanoids, rehabilitation and industrial automation.



Hendrik Wöhrle received his bachelor's and master's degrees in bioinformatics from the Freie Universität Berlin, Germany, in 2003 and 2006, respectively. He worked as a software developer of algorithms and signal processing methods from 2007 to 2009. Since 2009, he is with the Robotics Innovation Center (RIC) of the German Research Center for Artificial Intelligence (DFKI) in Bremen. In 2017, he received the Ph.D. degree (Dr.-Ing.) from the University of Bremen and is currently also a professor at University of Applied Sciences and Arts, Dortmund. His research interests are (biomedical) signal processing, machine learning, embedded systems, and reconfigurable computing.

José de Gea Fernández received his M.Sc. in Electronics Engineering (2002) from the Technical University of Catalunya (UPC), Spain and his Ph.D. in Robotics (2011) from the University of Bremen, Germany. Between 2003 and 2009 he was a Researcher at the Robotics Group of the University of Bremen. Since 2009 he is working at the Robotics Innovation Center of DFKI (German Research Center for Artificial Intelligence) in Bremen, where he currently leads the team Robot Control. He has co-authored over 45 scientific publications and acted as project leader in a number of German national and European projects. His research area is on mobile manipulation and collaborative robotics.

Andreas Müller obtained diploma degrees in mathematics (1997), electrical engineering (1998), and mechanical engineering (2000), and a Ph.D. in mechanics (2004). He received his habilitation in mechanics (2008) and is currently professor in robotics at the Johannes Kepler University, Linz, Austria. His current research interests include holistic modeling, model-based and optimal control of mechatronic systems, redundant robotic systems, parallel kinematic machines, biomechanics, and computational dynamics.

Frank Kirchner studied computer science and neurobiology at the University Bonn from 1989 until 1994, where he received his Dr. rer. nat. degree in computer science in 1999. Since 1994 he was Senior Scientist at the Gesellschaft für Mathematik und Datenverarbeitung (GMD) in Sankt Augustin and also a Senior Scientist at the Department for Electrical Engineering from 1998 at Northeastern University in Boston, MA, USA. In 1999, Dr. Kirchner first was appointed adjunct and then Tenure Track Assistant Professor at the Northeastern University, and since 2002 as a Full Professor at the University of Bremen. Since December 2005, Dr. Kirchner is also Director of the Robotics Innovation Center (RIC) in Bremen.