A Novel Concept for Building a Hyper-Redundant Chain Robot

KeJun Ning and Florentin Wörgötter

***Abstract*—This paper puts forward a novel design concept for building a 3-D hyper-redundant chain robot (HRCR) system, consisting of linked, identical modules and one base module. All the joints of this HRCR are passive and state controllable and share common inputs introduced by wire-driven control. The original prototype developed here, named 3D-Trunk, is used as a proof of concept. We will present its whole mechanical design and controller architecture. The key components of 3D-Trunk, its operational principles, and all implementation issues are exhibited and described in detail. Basic robotics analyses, dynamics simulations, and some experiments are also shown. This novel design concept is highly modular and scalable, no matter how many degrees of freedom are implemented and, thus, provides an affordable solution for constructing an HRCR.**

***Index Terms*—Hyper-redundant robot, mechanism design, robotics, wire-driven.**

I. Introduction

STUDYING PLANAR (2-D) or spatial (3-D) hyper-redundant chain robot (HRCR) systems is important, because such robots offer many independent degrees of freedom (DOF). Elephant trunk robots, snake robots, serpentine robots, etc, are representative cases [1]–[19]. Such robotic systems are designed to mimic their biological counterparts formally and functionally. The related literature is substantial, and many groups have built different HRCRs pursuing theoretical and application-driven research.

HRCRs are not easy to design, construct, and control. The most common approach to building an HRCR is by connecting several rigid links via an actuated revolute joint in a chain. Generally, many motors are distributed on the joints, and respective control systems are required. Another popular design utilizes parallel mechanisms to connect several links together [1]–[3]. Constructing compact robotic joints for an HRCR is the main mechanical design challenge because tradeoffs between different restrictions have to be considered [4]. Such robotic joints need to be highly maneuverable and strong enough to counteract the weight as well as the dynamic loads produced by their motions. A good review of the prior work had been presented in [4]. Some of the recent and representative works are discussed in the following.

Manuscript received January 9, 2009; revised June 12, 2009. First published October 30, 2009; current version published December 8, 2009. This paper was recommended for publication by Associate Editor H. R. Choi and Editor W. K. Chung upon evaluation of the reviewers' comments. This work was supported by PACO_PLUS from the European Commission and by the BMBF BCCN W3 project.

The authors are with the Bernstein Center for Computational Neuroscience, University of Göttingen, 37073 Göttingen, Germany (e-mail: nkj@sjtu.org, ning@bccn-goettingen.de; worgott@bccn-goettingen.de).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TRO.2009.2032968

The Robotics Institute at Carnegie Mellon University (CMU), Pittsburgh, PA, has developed a series of modular snake robots, e.g., [5] and [6]. Most of these snake robots are constructed as a conventional multijoint structure. The Institute also concentrated on the design of new joints for such robots. Shammas *et al.* improved Takanashi's angular swivel design [7] and presented new mechanical designs for compact 2-DOF [8] and 3-DOF [4] joint mechanisms. The basic mechanical components of their 3-DOF joint are two connected angular shafts. The useful DOFs (i.e., planar bending and orienting) are obtained by coordinating the rotation of these two shafts. Based on this novel 3-DOF joint design, they built a 12-DOF snake-like robot [4].

The Mobile Robotics Laboratory at the University of Michigan, Ann Arbor, developed two serpentine robots, called "OmniPede" and "OmniTread" [9]–[11]. "OmniPede" uses a motor at the end to rotate a "drive shaft spine," which provides mechanical power to each leg. "OmniTread" also utilizes a powered drive shaft spine to drive all treads of every segment. Their segments are all linked by 2-DOF pneumatically actuated joints.

Sujan *et al.* developed a lightweight hyper-redundant binary device BRAID [12], which was made of a serial chain of parallel stages. Each stage consisted of 3 DOF and was driven by three flexure-based legs, where shape memory alloy was employed to actuate them.

There exist some other interesting HRCRs, which offer new concepts to realize systems driven by fewer actuators thereby limiting the weight of the systems and possibly easing the control problem, as discussed in the following.

Paap *et al.* used cylindric rubber pieces to construct snake robot joints [13]. There were four motors mounted in a segment. The motors were used to wind and unwind wires to compress the serially connected cylindric rubber pieces in different directions for rotation. This solution enhanced the biological comparability of their snake robot.

Ananiev *et al.* presented a novel method for driving a hyper-redundant robot [14]. They used one irreversible motor to drive a flexible shaft for transporting power. All the multiple modules of the robot utilized several clutches to selectively distribute the torque and rotation of the flexible shaft independently to any of the robot modules [14].

Researchers at Clemson University, Clemson, SC, developed an elephant trunk robot [15], which features a backbone of 32 DOF, with eight independently driven components and 16 "disks" consecutively connected by 2-DOF universal joints. Of the possible 32 DOF, the hybrid cable and spring servo system creates a manipulator with 8 DOF that are user controllable and 24 DOF that are coupled to the controllable DOFs [15], thereby adjusting to environmental constraints on contact.

Ohno and Hirose presented a slime robot module [16]. Each module owns 3 DOF, with pneumatic actuators, valves, sensors, and a microprocessor in its body. By serially connecting these modules, they built a Slim Slime Robot [16].

Inspired by the elephant-trunk solution [15], the Autonomous Systems Laboratory of the Swiss Federal Institute of Technology, Lausanne, Switzerland, presented the development of a bio-mimetic spine for the humanoid robot called Robota [17]. This spine is composed of four vertebrae parts, linked through spherical bearings. There are distributed hydraulic pumps and springs to drive the vertebrae and bend the spine.

In summary, the traditional distributed actuation approach lets an HRCR realize flexible and dexterous motion by coordinating several joints simultaneously, but it is difficult to design and control. As discussed earlier, some specific solutions [13]–[16] had been presented for novel joint design and power input. Such efforts are all trying to implement complex HRCRs by overcoming critical size and weight restrictions. Many times this, however, reduces manipulability or dexterousness. This is due to the fact that in these designs, some degree of coupling exists for sharing the input power. Here we present a novel solution for building an HRCR, which consists of passive and state controllable joints and shares a common driving input introduced by wire-driven control. The prototype developed here, named 3D-Trunk, offers several advantages compared with related designs [13]–[16] and is used to verify the feasibility and maneuverability of the novel concepts and design principles.

The remainder of this article is organized as follows. In Section II of this paper, the concept is presented. The original prototype design and some implementation issues are described in detail in Section III. Section IV describes the distributed control architecture of our prototype and some issues related to electronics. In Section V, we analyze the robot and show some simulations. Experimental results of our 3D-Trunk prototype are provided in Section VI. Finally, conclusions and application potential are presented in Section VII.

II. Concept of the Novel Hyper-Redundant Chain Robot

A. Passive Joint with State Control

Fig. 1(a) shows the general 3-D hyper-redundant chain mechanism schematics. It consists of a set of links serially connected by revolute joints in a chain. Each revolute joint has 1 DOF. The angle between two neighboring joint axes can be arbitrary, but in our design, we use perpendicular axes, as control would otherwise become more complicated.

In the case shown in Fig. 1(a), we assume that all the joints of the whole chain mechanism are passive. If we lock some of the revolute joints, the locked joints will not be rotatable, and the DOF of the whole chain will essentially become degenerated. In Fig. 1(a), we locked joint $i-1$ and $i+1$ (by control input EN_θ_{i-1} and EN_θ_{i+1}), only joint i is in the unlocked state. As a consequence, the original 3 DOF have degenerated to 1 DOF, as shown in Fig. 1(b). Because all the joints were passive, any external force or torque will rotate the unlocked joint i.

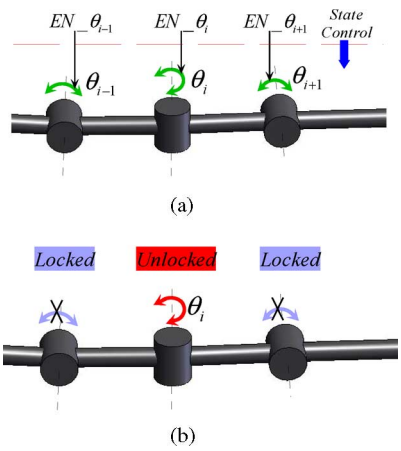


Fig. 1. Three-dimensional hyper-redundant serial-link mechanism schematics. All the revolute joints are passive and with binary-state control. They can only be in locked or unlocked states.

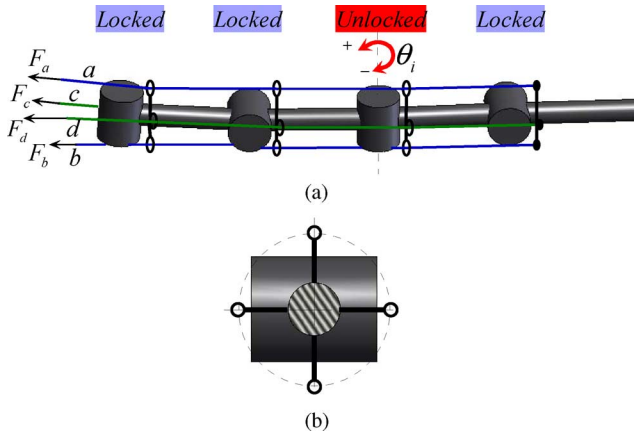


Fig. 2. All the joints of the chain share the same external drive. There are two pairs of driving-wires that actuate the whole passive chain.

With this concept, we can control any one joint's motion step-by-step; then by altering the loading configuration, we can change the whole chain's shape according to our expectation. The dynamics and kinematics characteristic of the *n*-DOF chain will depend on the $\mathrm{EN}\_\theta_i(t)$, $(i = 1, 2, \ldots, n)$ scheduling and "external torque" input sequence.

In general, our design relates to that presented by Ananiev *et al.* [14] but differs in many ways as discussed in the following.

B. DOF State Controllable and Driving Shared

Similar to [20]–[25], we also use light flexible wires to control our system because wire-driven systems have many advantages, such as a simple structure, large workspace, high speed, and low cost. Thus, wire-driven is an easy and effective approach to drive our passive hyper-redundant serial kinematic chain. All the passive joints of the whole chain share the driving effect introduced by the wires, which need to be kept in a taut state, because wires can only impose unidirectional constraints (only pull, not push).

Fig. 2(a) shows a simplified implementation scheme. There are two pairs of driving-wires (*a*, *b* and *c, d*), which drive a

hyper-redundant serial kinematic chain. The neighboring revolute joints of this chain are perpendicular to each other, designed based on the concept mentioned earlier. As shown in Fig. 2(b), there are four "rings" symmetrically distributed and fixed at each link. These rings are used for turning and for guiding the wires. Two opposite rings are coplanar with the joint's axis, and the other two are perpendicular to them. These rings are located very close to the joint. As shown in Fig. 2(a), two pairs of wires are routed along the chain via many such rings. For each link, each pair of driving-wires is routed via all ring pairs on opposite sides. Thus, similar to muscles and tendons, this design focuses on a pulling-only actuation, where wires on opposite sides provide the required opposition (flexor and extensor) in the actuation. Such flexible wires make the mechanical design and its control easier than when using alternative (stiff) push–pull mechanisms.

For this kind of HRCR, every unlocked joint is differentially driven by the wire pairs using windlasses. In the situation shown in Fig. 2(a), only joint *i* is unlocked, the wires' traction will not affect the locked joints. If wire *c* is wound and wire *d* is unwound, joint *i* will rotate positively and *vice versa*. The pulling caused by wires *a* and *b* will have very limited effect on joint *i*, due to the torque they generate, which is mostly perpendicular to the axis of joint *i*. Even this weak effect could be eliminated totally by additional mechanical design solutions. As this effect is small, we have abstained from this, though. Therefore, for the unlocked of joint *i*, wires *c* and *d* are the effective pair of wires.

For joint *i*, the resulting driving torque is

$$T_i = F_c r_{ic} - F_d r_{id}. \tag{1}$$

Here, T_i is the effective driving torque of joint *i*, and F_c and F_d are the tensile forces of the wires *c* and *d*, respectively. Finally, r_{ic} and r_{id} are the arm lengths for F_c and F_d and depend on θ_i, the rings' implementation and joints' mechanical design.

Theoretically, one pair of driving-wires would be enough to drive the whole chain. This would, however, require allowing the wires to be twisted along the chain, which is mechanically infeasible. By contrast, arranging the revolute joints perpendicular one by one is quite a useful configuration, easing design and kinematics analysis. These are the reasons why two pairs of driving-wires are employed in our system.

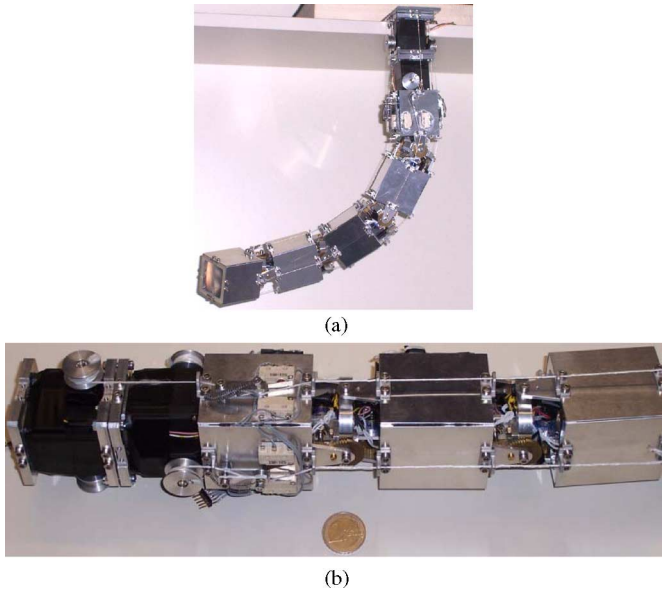


Fig. 3. Original 3D-Trunk prototype. (a) 3D-Trunk is an 8-DOF HRCR system. (b) Key structure of this prototype.

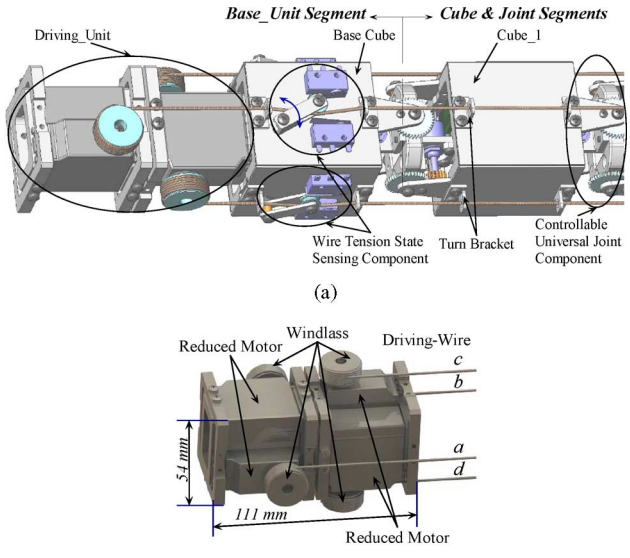


Fig. 4. 3D-Trunk is a wire-driven system. This HRCR comprises a Base_Unit, many identical modularized Cubes, and Controllable Universal Joint Components. (a) Overview of a 3D-Trunk. (b) Compact Driving_Unit. All the joints share this common driving input.

III. 3D-Trunk: A Design Paradigm Based on This Concept

A. Introduction of 3D-Trunk's Mechanical Design

Fig. 3 shows our original prototype system called "3D-Trunk." It is an 8-DOF wire-driven system, constructed by the concept presented. All electronic components and microcontrollers were embedded inside.

Fig. 4 shows the key implementation details of 3D-Trunk. Based on the segments' function partition, the chain consists of a "Base_Unit" and many identical modularized "Cube & Joint" segments.

The Base_Unit is the power segment of the whole chain for housing four Reduced Motors (with reduction gears), as well as the motor control boards. At its one end, the Base_Unit interfaces with the first Cube by a joint. At the other end, the Base_Unit provides the interface components for mechanical mounting to some other structures. The Base_Unit consists of a Base Cube and a Driving_Unit [see Fig. 4(a)].

Fig. 4(b) shows the design of the compact Driving_Unit. There are four Windlasses, actuated by four DC-Reduced Motors and driving four Driving-Wires. In fact, these four Reduced Motors form the two independent wire-driven pairs (*a*, *b* and *c*, *d*), and all the DOFs of this system share these two differential inputs. Each pair of Driving-Wires needs to be simultaneously coordinated. The Base Cube houses the required control boards.

For the original prototype, the Reduced Motors were reconfigured starting from off-the-shelf servos. They are driven by the new circuit boards we developed. In this way, we can achieve a suitable actuation performance. At present, each Reduced Motor can generate a maximal torque near 1 N·m and a maximal rotational speed (without load) of approximately 20 r/min.

As shown in Fig. 4(a), in Cube & Joint Segments, the same cubes and joints components are linked with each other. The Cubes are used for positioning the joints, for housing the distributed electronics boards, and for routing cables inside. They are the movable segments of our HRCR.

As shown in Fig. 4(a), there are Turn Brackets mounted outside the cubes (Base Cube, Cube_1, . . . , Cube_*n*). They work as simplified pulleys for positioning and guiding the Driving-Wires (the same function as the rings shown in [see Fig. 2(a)].

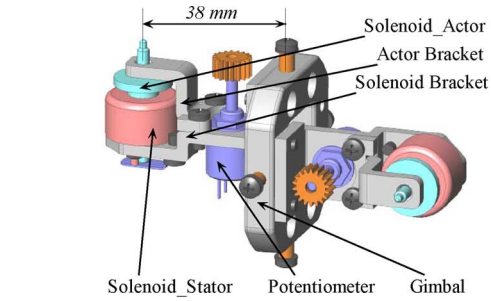
B. Design of 3D-Trunk's Controllable Universal Joint Component

As mentioned in Section I, for constructing practical hyper-redundant robots, the joint is the main mechanical design challenge [4]. Generally speaking, a clutch is a suitable device to control the revolute joint's binary state (locked and unlocked). However, off-the-shelf products are not suitable for our requirements. If we constructed the clutch for a joint based on controlled friction, the required structure would be rather heavy, taking up much space, while the available stable locking force would still be too limited. This is due to friction being proportional to the normal contact force. Thus, a different, compact, and strong enough binary-state locking component is the key to implementing our new concept.

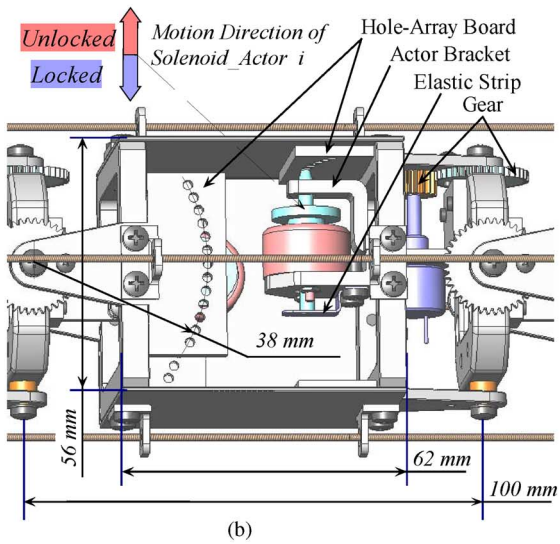
Fig. 5 discloses our solution for constructing an effective and compact "clutch" for this HRCR. This design achieves much higher locking torque in a very compact and light implementation and provides joint angle feedback information.

The key device is called "Controllable Universal Joint Component" (CUJC), which is shown in Fig. 4(a). As shown in Fig. 5(a), a CUJC contains a Gimbal, whose two perpendicular axes coincide at each joint. Some key parts of this design are mounted on the Gimbal. One CUJC is used to bridge two neighboring Cubes to provide 2-way independent clutches and to house 2-way joint angle sensors. Fig. 5(b) and (c) disclose 3D-Trunk's inner mechanism and the assembly design of the Cube and the CUJC in detail.

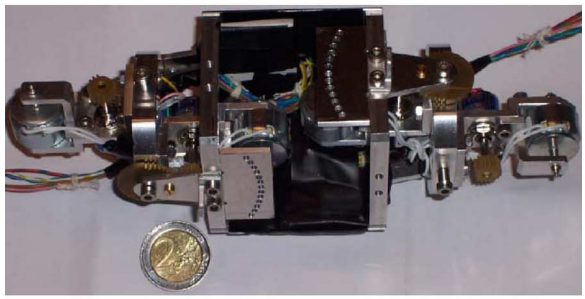
As shown in Fig. 5, a linear solenoid (pull type) driving mechanism was employed to construct a practical and compact binary-state clutch for 3D-Trunk. Linear solenoids are ideal for high force, short stroke applications. The moveable iron core of the solenoid [see the Solenoid_Actor in Fig. 5(a) and (b)] is used to lock and unlock its joint via a Hole-Array Board, which provides arrayed holes to plug into. The Solenoid_Actor has a pointy tip for easier plugging into the holes. In Fig. 5(a) and (b), the geometrical arrangement of this clutch is shown. The Solenoid_Stator encapsulates a coil and sliding bearing, which is mounted on the Solenoid Bracket. These are the key parts of the controllable binary-state locking mechanism of this design.



(a)



(b)



(c)

Fig. 5. Mechanical design of 3D-Trunk's CUJC and Cube. (a) A CUJC offers 2 DOF. (b) Inner mechanism of the Cube and CUJC. (c) Completed Cube and CUJC.

For each Hole-Array Board, the holes are circularly and evenly distributed [see Fig. 5(b)]. The angle space between two neighboring holes depends on the diameter of the "pin," the radius of the hole-array, strength of material, components' size constraints, etc. For our original prototype, one joint's working range is from -27° to $+27^\circ$, with a resolution of 4.5°. While spatial resolution of this design is somewhat limited, those parameters have been chosen to assure stable and robust performance of our prototype.

In Fig. 5(b), if the solenoid is not powered, the Elastic Strip will push the Solenoid_Actor into a plug. If it is coaxial with a hole in the Hole-Array Board, then the plugging action will immediately be successful. If not, the joint will continue to move until a neighboring hole is met to allow locking. As a

consequence, the discrete resolution of the joint of HRCR is determined by the design of the Hole-Array Board.

In this design, pull-type linear solenoids were employed and the joint's locked state is identical with solenoid's unpowered state. In this way, power consumption of the whole HRCR is greatly reduced.

As shown in Fig. 5(b), if the solenoid is powered, the Solenoid_Stator will pull the Solenoid_Actor by an electromagnetic force. If this effect is strong enough, the Solenoid_Actor will be unplugged from the Hole-Array Board. The Actor Bracket is used to position the Solenoid_Actor (together with the sliding bearing of Solenoid_Stator) and to enhance the locking stiffness of the CUJC. It also improves the stress distribution and reduces friction between the sliding bearing of Solenoid_Stator and Solenoid_Actor.

In order to increase the reliability of a locking action, the Elastic Strip needs to be stiff enough. Sequentially, the solenoid needs to offer enough pulling force to counteract the elasticity of the Elastic Strip, the friction between its stator and actor, and the friction between the actor and the plugged hole of the Hole-Array Board.

For increasing the maximum pulling force, a pulse-width modulation (PWM) controlled current is used to power the solenoid. By powering the solenoid with higher voltage and low duty cycle, we can obtain much higher pulling force while the solenoid will still not burn out.

As shown in Fig. 5(a) and (b), this solution can also provide joint angle feedback information. A potentiometer is used to sense a joint's absolute angle. A pair of gears is employed to transmit motion. As the bigger gear is fixed to the Cube, the obtained reduction ratio ($i = 9/20$) increases the potentiometer's effective resolution. Small plastic conductive potentiometers were employed in our prototype for getting low electrical noise, high linearity, and long life. The angle feedback is used not only to measure a joint's real rotation but to deduce plugging positions or check the result of a locking (plugging) action as well.

This type of "clutch" implementation is compact and can achieve much higher locking torque than a friction-based approach, because the available shearing force between "pin" and "hole" is much stronger, depending on the strength parameters of the employed material. The current locking mechanism design provides 7 N·m as its allowable maximal locking torque.

This solution, however, also has some disadvantages. The pin's plugging and unplugging actions are relatively slow, and the locking angle steps of each joint are discrete and predetermined.

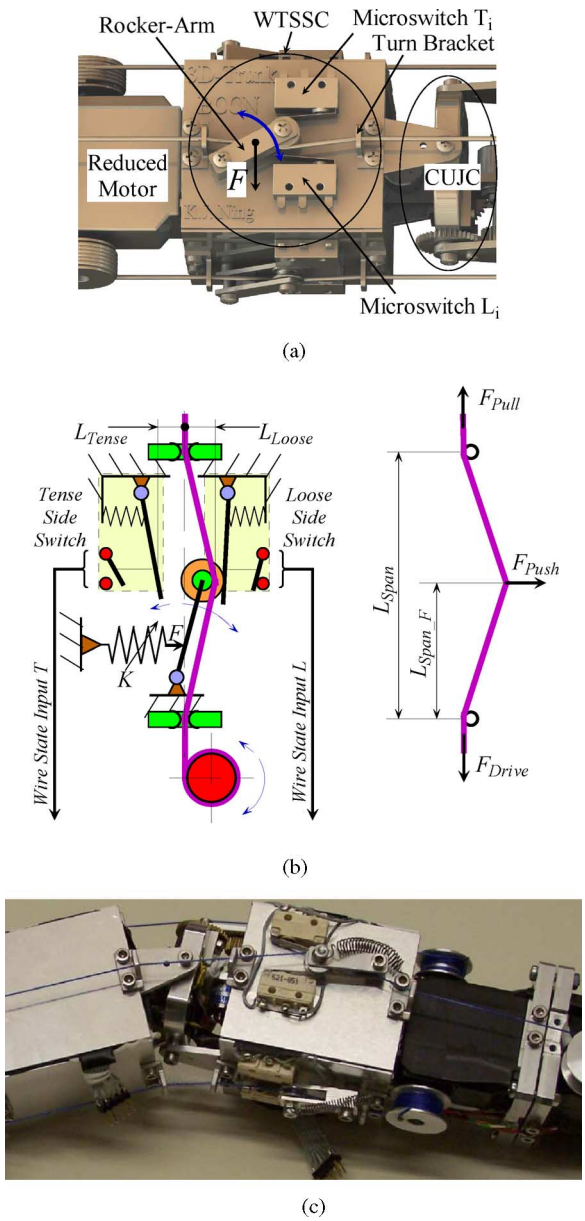


Fig. 6. Design and principle of the WTSSC. (a) WTSSC's design. One Rocker-Arm and two arranged Microswitches are used to sense a Driving-Wire's tension state. (b) Principle of WTSSC. This is a discrete solution to deduce the pulling force of a wire, with resilience capability. (c) Completed WTSSCs and Base Cube.

C. Design of 3D-Trunk's Wire Tension State Sensing Component

3D-Trunk is a wire-driven system. During a joint's motion period, the actuation of all Driving-Wires needs to be simultaneously coordinated. Thus, sensing of the Driving-Wires' states is important.

Similar to the case discussed in (1) and the situation shown in Fig. 2(a), for joint i, wires c and d are its effective pair of wires. In this case, to achieve a positive rotation of joint i, wire c is wound tighter for a positive contribution. If wire d is at the same time unwound too slowly, it will be too tight and thereby counteract some power from wire c; if unwound too fast, wire d will be too loose and may produce problems at its windlass (see Fig. 4, e.g., wire d may hop out from its windlass). Furthermore, during this action, tensions of the nonused pair of wires, i.e., a and b, will vary, too. This is due to the resulting geometrical configuration of the chain [see Figs. 4(a) and 6(c)]. This unwanted length change also needs to be compensated. Note, calculating the Driving-Wires' length changes to control windlass rotation is highly error-prone, as the

geometrical situations, which occur during trunk motion, are quite complicated with complex friction changes, etc. Our solution, which will be discussed in the following, provides feedback information of the driving wires' tension states, as well as some resilience for self-adjusting and overcoming the nonlinear coupling existing in this wire-driven system.

As shown in Fig. 4(a), there are four Wire Tension State Sensing Components (WTSSC) symmetrically distributed on the four outer faces of the Base Cube. The design of the WTSSC is shown in Fig. 6(a). For a WTSSC, a rotatable Rocker-Arm is pulled by a spring and triggers a pair of Microswitches. Each Driving-Wire is routed along a small pulley inside of the Rocker-Arm.

Fig. 6(b) shows the acquisition principle of the wire-tension state. If the wire is too loose, the spring will push the Rocker-Arm to press the Loose-Side Switch, whereas the Tense-Side Switch will be depressed and *vice versa*. By adjusting L_{Loose} and L_{Tense}, the spring's elastic coefficient K, and L_{Span} and $L_{\mathrm{Span\_F}}$, we can obtain different state-threshold definitions. The actual pulling force from the Driving-Wire can also be deduced. In fact, this is a discrete solution for deducing the pulling force of a wire, with resilience capability. If a rotation sensor was employed here, we could obtain more accurate results but would need more controller resources (e.g., analog/digital (A/D) converter port). On the other hand, the solution presented in Fig. 6 is simple and very practical.

In addition to tense and loose state, there is also a third, midstate. As shown in Fig. 6(b), if L_{Loose} plus L_{Tense} are bigger than the external diameter of the pulley (D_{Pulley}) mounted on the Rocker-Arm, a middle domain exists where none of the two Microswitches will be pressed. In fact, this third state is vital for representing the wire's not-loose and not-tense state. Its range is $L_{\mathrm{Loose}} + L_{\mathrm{Tense}} - D_{\mathrm{Pulley}}$. Note L_{Loose} and K [see Fig. 6(b)] define a threshold F_{NL}, where the WTSSC gets into midstate if the pulling force F_{Pull} is larger than F_{NL}. This threshold is important for the control of our HRCR, as will be discussed in the following.

These parameters and state definitions are important for closed-loop control of our wire-driven system. By gathering the WTSSC outputs together with those from the potentiometers [mounted on CUJCs for offering joint angle information, see Fig. 5(a) and (b)] in real time, the rotation of the unlocked joint of 3D-Trunk becomes controllable.

For example, similar to the case discussed in (1) and the situation shown in Fig. 2(a), to achieve a positive rotation of joint i, wire c can be wound by its motor utilizing various driving strategies, and its WTSSC will always be in a tense state. The motors for driving wires d, a, and b need to be self-adjusting to keep their respective WTSSC's state changing between loose and midstate (to keep the pulling force at approximately F_{NL}). This is the easiest way to ensure the stability of this wire-driven robot (overcoming the coupling problems of this nonlinear system) and to reduce the power dissipation.

So far, the key mechanisms of our novel wire-driven HRCR have been described in detail. These solutions are the foundation of 3D-Trunk's embedded controllers and its application.

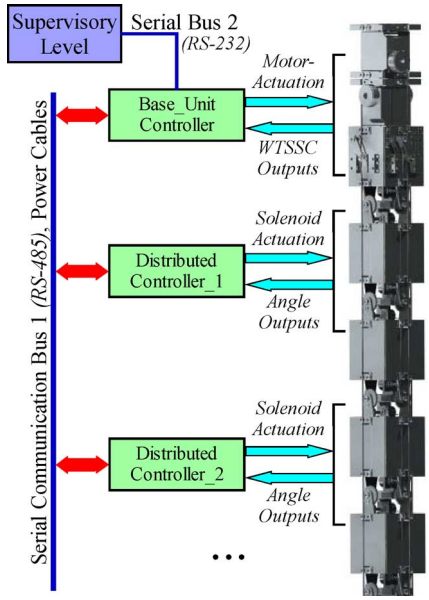


Fig. 7. Distributed control architecture scheme for HRCR.

IV. Controller Architecture of 3D-Trunk

A distributed control architecture is a suitable approach for this HRCR (see Fig. 7).

For 3D-Trunk, all power, driving cables, and communication twisted pairs were routed along and inside the Cubes (see Fig. 3). Communication between the Base_Unit Controller and the Distributed Controllers takes place via an RS-485 serial bus. The RS-485 was employed because of its high immunity to noise and ability to drive large distances with high data rates. Solenoids are powered by PWM current, and each peak current is about 1.6 A. Thus, they could be strong interference sources. Since the RS-485 is differential, however, it resists electromagnetic interference from distributed solenoids and driving cables. Thus, this bus can be used for building very long HRCRs.

We developed specific embedded control boards for 3D-Trunk. These modularized boards are distributed and embedded in some Cubes of the whole HRCR. Each entire Distributed Controller consists of a microcontroller board and a MOSFET-array board. Each owns a unique ID. The Base_Unit Controller consists of a microcontroller board and two motor-driving boards. Such a separate arrangement reduces interference and houses them better in their Cubes.

The Distributed Controller modules are based on an ATMEGA16 microcontroller (AVR core from ATMEL). Counting in the issues of cable routing and maintenance, each Distributed Controller module of 3D-Trunk is in charge of two neighboring Cubes (in other words, 4 DOF), as shown in Fig. 7. The potentiometers of the joints were connected via the microcontroller's A/D converter port. The internal peripheral timers of ATMEGA16 generate the PWM signals (for the solenoids), and the programmable Input/Output ports drive a MOSFET-array board to actuate the solenoids.

The Base_Unit Controller module is encased in the Base Cube. It is in charge of controlling the four motors

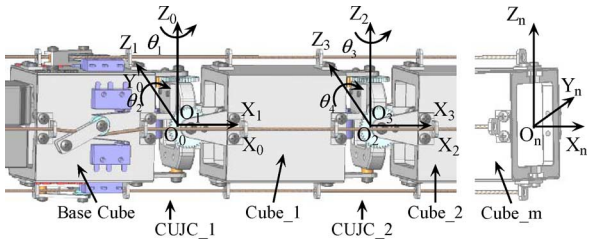


Fig. 8. Coordinate frames design of an *N*-DOF 3D-Trunk based on D–H parameters.

synchronously, interfacing the WTSSCs, etc. Furthermore, it is the master node of the Communication Bus 1 shown in Fig. 7. The Base_Unit Controller communicates with the Distributed Controllers and coordinates their actions.

We can access and operate the whole HRCR by connecting a supervisory level controller. A command set was implemented in the firmware of the Base_Unit Controller. The supervisory level controller sends some special command packets to 3D-Trunk; 3D-Trunk will understand them and operate accordingly. In this way, a user can operate this system easily without prior knowledge of the low-level implementation details of the whole mechatronic system.

At present, there are some autonomous functions implemented in 3D-Trunk's embedded controllers; for example, overtime protection, self-triggering to adjust all Driving-Wires' tension states, autonomous escaping from a possible "stuck" situation (if an actuated Solenoid_Actor fails to escape from the Hole-Array Board, the embedded controller will know and behave intelligently to solve this problem immediately), self-test, etc.

The system level design of 3D-Trunk offers a high potential for extensions. We can mount various sensors and mechanical effectors and port their signals to the remaining microcontroller resources for further research and applications.

V. Kinematics and Dynamics of 3D-Trunk

In this section, we present the computational model for the HRCR. The whole chain is described with the standard Denavit–Hartenberg (D–H) parameters.

A. Coordinate Frames Design and Mass Properties

Fig. 8 shows the coordinate frames design of an *N*-DOF 3D-Trunk. The related D–H parameters of our original prototype are listed in Table I. Due to the concepts presented in this paper and the highly modularized design of the HRCR, it is easy to extend and model such a robot with an arbitrary number of DOF.

In the case shown in Fig. 8, all Cubes, CUJCs, and the one-end cube (Cube_m) are assembled in the same direction. In this situation, we only need to model and calculate each different component once, and then, we can deduce the whole chain's mass, distributions and inertia matrixes to conduct a dynamics analysis. This accurate dynamics model is the key to our system specification, part checking, controller design, and performance improvement.

TABLE I
D–H Parameters of 3D-Trunk

Link	α_i (Rad)	a_i (mm)	d_i (mm)	θ_i (Rad)
1	$-\pi/2$	0	0	θ_1
2	$\pi/2$	100	0	θ_2
3	$-\pi/2$	0	0	θ_3
4	$\pi/2$	100	0	θ_4
…	…	…	…	…
N	$\pi/2$	81	0	θ_n

TABLE II
Mass Properties of Components

Name of Link	Mass (g)	Center of Mass (mm)	Moments of Inertia ($g\times mm^2$)
CUJC	130	x = 0 y = -2.42 z = -2.42	$I_{xx} = 2.15\times10^4$ $I_{yy} = 1.05\times10^5$ $I_{zz} = 1.05\times10^5$ $I_{xy} = I_{yx} = 8.80\times10^3$ $I_{yz} = I_{zy} = -7.74\times10^2$ $I_{xz} = I_{zx} = -8.80\times10^3$
Cube	119	x = -50 y = -0.34 z = 0.34	$I_{xx} = 1.00\times10^5$ $I_{yy} = 1.32\times10^5$ $I_{zz} = 1.32\times10^5$ $I_{xy} = I_{yx} = 1.95\times10^3$ $I_{yz} = I_{zy} = 1.34\times10^1$ $I_{xz} = I_{zx} = 1.95\times10^3$
Cube_m	103	x = -35.4 y = -0.39 z = 0	$I_{xx} = 8.87\times10^4$ $I_{yy} = 1.04\times10^5$ $I_{zz} = 1.13\times10^5$ $I_{xy} = I_{yx} = 1.77\times10^3$ $I_{yz} = I_{zy} = 0$ $I_{xz} = I_{zx} = -1.00\times10^{-2}$

The detailed mass properties of 3D-Trunk's components are listed in Table II (not including cables and controller boards), calculated in the respective coordinate frames (see Fig. 8).

B. Inverse Dynamics Simulation of the HRCR

The Robotics Toolbox V7.1 [26] and MATLAB were used to test the validity of the coordinate frames design, as well as to conduct analyses on kinematics and dynamics.

Let us define the vector of joint variables for this *N*-DOF HRCR (see Fig. 8) by

$$\mathbf{q} = [\theta_1, \theta_2, \ldots, \theta_N]^T . \quad (2)$$

The equations of motion for an *N*-DOF serial robot are given by [26]

$$\mathbf{T} = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{F}(\dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q}) \tag{3}$$

where $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ are, respectively, the $N \times 1$ joint velocity and acceleration vectors, $\mathbf{T}$ is the $N \times 1$ input torque vector, $\mathbf{M}(\mathbf{q})$ is the inertia matrix, $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})$ is a vector resulting from Coriolis and centripetal accelerations, $\mathbf{F}(\dot{\mathbf{q}})$ describes viscous and Coulomb friction, and $\mathbf{G}(\mathbf{q})$ is the generalized gravitational force vector [26]. Note that we did not consider $\mathbf{F}(\dot{\mathbf{q}})$ here.

In fact, this kind of HRCR-dynamics modeling is the same as that used for traditional serial robots, because they are all multilink rigid systems. By inverse dynamics, we can compute the required joint actuator torques (forces) from a given trajectory of the manipulator.

For 3D-Trunk's inertia matrices, only the CUJC's mass property can possibly vary within its own coordinate frame. This is caused by the Solenoid_Actor's stroke (state-shifting from locking to unlocking; see Fig. 5[b]). In our design, the employed solenoid is small. The Solenoid_Actor weighs 5.5 g, and its stroke length is 1 mm. Therefore, the CUJC's mass distribution variation during a stroke is very limited, and we can ignore it. Thus, in the following dynamics analysis, we use the mass properties listed in Table II during the whole process.

Given a planned trajectory, a traditional serial robot can control its distributed actuators (e.g., motors) synchronously and the required torques of these joints are calculated by inverse dynamics. For the HRCR presented in this paper, the joints can only be unlocked asynchronously and share the driving from the external Driving-Wires. Its operation depends, thus, on the used driving sequence. Furthermore, the interpretation of some calculation results will be different.

1) For the one unlocked joint, the calculated torque arises from the Driving-Wires.
2) For all other (locked) joints, present joint angles are kept and torques are provided by the locking parts (e.g., the Hole-Array Board and Solenoid as shown in Fig. 5[b]).

So that we get for the input torque vector $\mathbf{T} = [T_1, T_2, \ldots, T_N]^T$

$$T_i = T_{Wi} + T_{Li}, \qquad i = 1,\ 2,\ ...,\ N \tag{4}$$

where T_{Wi} is the torque generated by the routed Driving-Wires, and T_{Li} is the torque generated by the routed locking mechanism (see Fig. 2(a)).

Given joint *j* is unlocked, then for all joints of this HRCR

$$\begin{cases} T_{Li} = 0, & i = j \\ T_{Li} \neq 0, & i \neq j \end{cases} \tag{5}$$

where

$$T_{Wj} = T_j,\ j \text{ is the unlocked joint} \tag{6}$$

and

$$T_{Li} = T_i - T_{Wi} \qquad (i = 1,\ 2,\ \ldots\ N \text{ and } i \neq j) \tag{7}$$

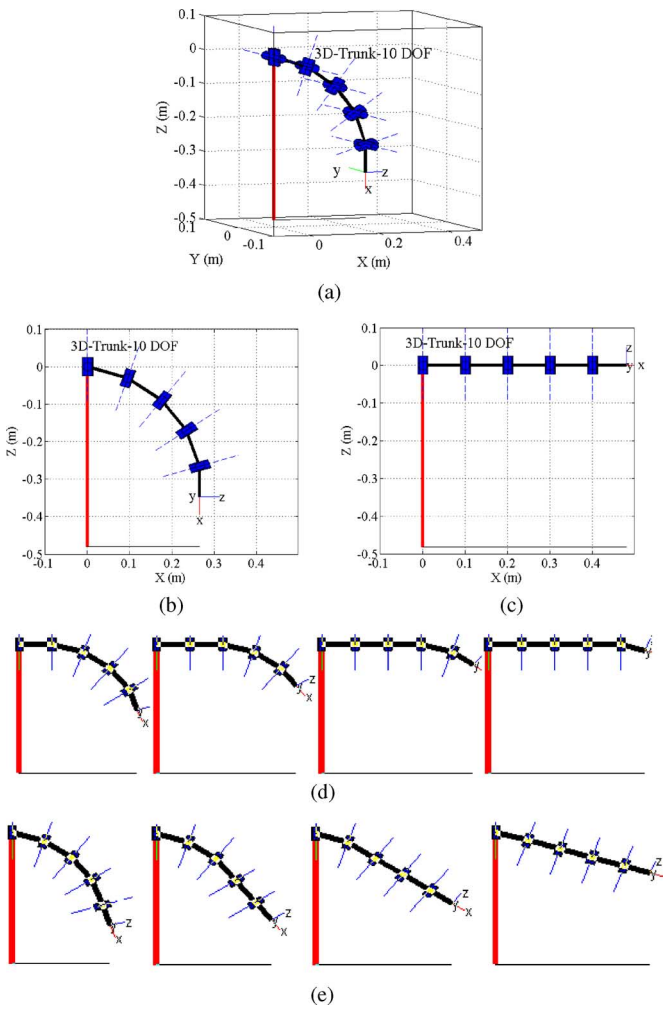


Fig. 9. Poses of a simulation experiment. (a) Visualization of a 10-DOF HRCR. (b) Start pose. (c) Stop pose. (d) Motion sequence from J_2 to J_{10}. (e) Motion sequence from J_{10} to J_2.

Equation (6) presents the approach to control these Driving-Wires. For this, the outputs of WTSSCs are used (see Section III-C). Equation (7) discloses how to calculate the load for checking the original design of the locking mechanism.

Fig. 9 shows a 10-DOF HRCR and two poses: start and stop. The corresponding joint space coordinates are as follows:

$$\mathbf{q}_{\text{start}} = [\theta_1, \theta_2, \ldots, \theta_{10}]^T = [0, \pi/10, 0, \pi/10, \ldots, 0, \pi/10]^T_{1\times 10}$$
$$\mathbf{q}_{\text{stop}} = [0, 0, \ldots, 0]^T_{1\times 10}.$$

This is a simulation for lifting the whole chain from a downward pose to a horizontal pose in a vertical plane. During this course, the Driving-Wires powered by the motors only need to work against gravity and against the dynamic load of the chain robot.

The five even-numbered and parallel joints (J_2, J_4, J_6, J_8, and J_{10}) of the 10-DOF HRCR have to be operated one by one. The whole running time was 2.5 s and was evenly divided into five time slices. In the following two experiments, we use two different driving sequences.

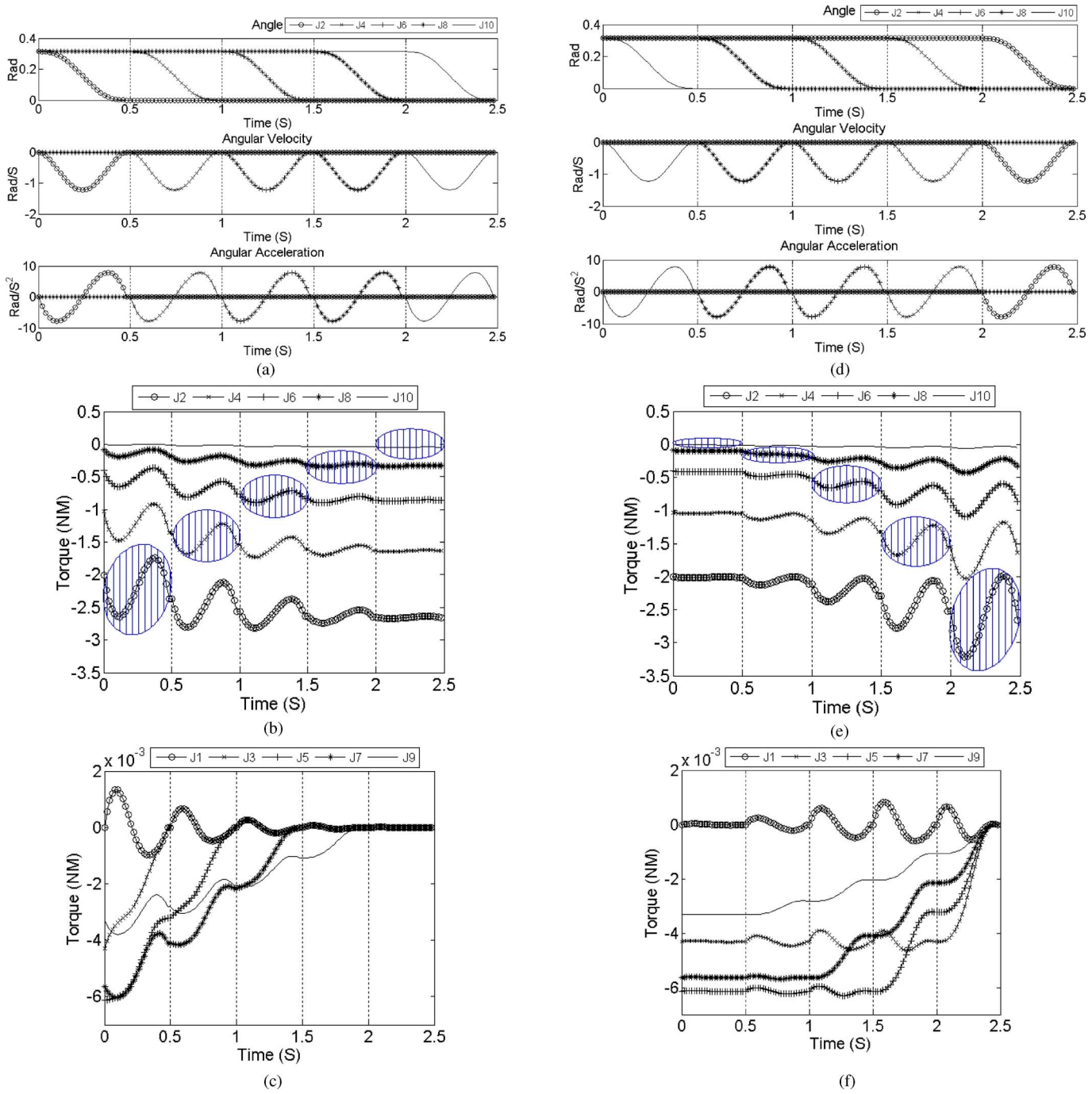


Fig. 10. Comparison simulation for an HRCR, with different driving sequences. (a)–(c) show results for case 1 [see Fig. 9(d)], (d)–(f) those for case 2 [see Fig. 9(e)]. Parts (a) and (d) are the joint trajectories, velocities, and accelerations. For (b) and (e), at any moment, only one joint is marked when it is actively actuated by the Driving-Wires, and at the same time, the remaining joints are passively "actuated" by the locking mechanism, respectively. (c) and (f) show torques of the vertical joints, which were held by their respective locking mechanisms.

Case 1: As shown in Fig. 9(d), the driving sequence is from base to end, one by one

$$J_2 \rightarrow J_4 \rightarrow J_6 \rightarrow J_8 \rightarrow J_{10}.$$

Case 2: As shown in Fig. 9(e), it is from end to base, one by one

$$J_{10} \rightarrow J_8 \rightarrow J_6 \rightarrow J_4 \rightarrow J_2.$$

Here, $J_i (i = 2, 4, 6, 8, 10)$ means that only joint i is in an unlocked state and that the other joints are all locked.

Fig. 10(a)–(c) show results for case 1 and Fig. 10(d)–(f) those for case 2. For these five time slices, the joint space trajectories of the unlocked joint were planned using a fifth-order polynomial from the Robotics Toolbox [26]. Fig. 10(a) and (d) show the corresponding kinematics information (joints' trajectories, velocities, and accelerations) in joint space. Visualizations of

these processes are shown in Fig. 9(d) and (e), respectively. Odd-numbered joints were all locked and their data are not plotted.

Using the joint space trajectories, the required joint torques can be computed by inverse dynamics computation. The required torques of the five even-numbered joints are plotted in Fig. 10(b) and (e). Note that any time during the operation, only one joint was driven by a pair of motors, and the others were held by their own locking mechanisms [see Fig. 9(d) and (e)].

As shown in Fig. 10(b) and (e), at any moment along the time axis, only one curve is marked. The marked zone means that only this joint is in an unlocked state and driven by the external driving-wires [see (6)].

Fig. 10(c) and (f) show the torques of the locked, odd-numbered joints, which are quite small, because their centers of mass are very close to the symmetry plane of the prototype (in Fig. 8, it is the fixed plane $X_0O_0Z_0$). They were also held by their respective locking mechanisms. Furthermore, these torques will become zero in the end, because the axes of the odd-numbered joints are vertical at the stop pose [see Fig. 9(c)]. The results shown in Fig. 10(c) and (f) are helpful to understand the whole process and to verify our calculations.

The redundancy of chain robots allows obtaining the same configuration by different sequences, some using low and some high forces (torques). In our design, once locked, a joint is power-free. This allows optimizing shape changes by choosing low-force sequences. In this way, the HRCR can be employed in an energy-saving domain using smaller motors as compared with systems that need to provide continuous holding power. Common clams and mussels employ a similar energy-saving principle as their muscles, which are different from all other animals, can "crystallize" in their holding mode (when the clam is closed) [27], thereby no longer having to provide costly continuous contraction forces.

VI. Experiments

Two results of our 3D-Trunk prototype are shown in Fig. 11(a) and (b). These demonstrations are used to verify the feasibility and maneuverability of this novel concept and design.

During the experiment shown in Fig. 11(b), the peak current summation (all four motors) was 0.8 A at 5 V. Thus, this kind of HRCR has the potential to use small motors to drive the whole system, in contrast to the commonly used distributed active driving approach.

Fig. 11(c) shows experiments for testing the locking mechanism of the 8-DOF robot. On the right-hand side subfigure, the locking mechanism close to the Base Cube provided a torque of about 7.1 N·m, and the shearing force between the CUJC and the Hole-Array Board was about 187 N. It is difficult to obtain such strong locking forces with conventional clutches of the same size and weight like our solenoid-based lock. Thus, our design provides a quite light, compact, and strong solution for building an HRCR.

Fig. 12 shows the real-time response curves of two joints. For these measurements, the employed on-line sampling frequency is 20 Hz, and the sampling time is 4 s. They are from the movement sequence shown in Fig. 11(b). Once the unlocked joint is close enough to the target position (defined by an error band parameter), the embedded controllers will command the locking mechanism to lock this joint immediately. We can see that, after the locking action, the joint-angle is very stable.

These experiments also disclosed some problems. The present controllable binary-state locking mechanism has exhibited

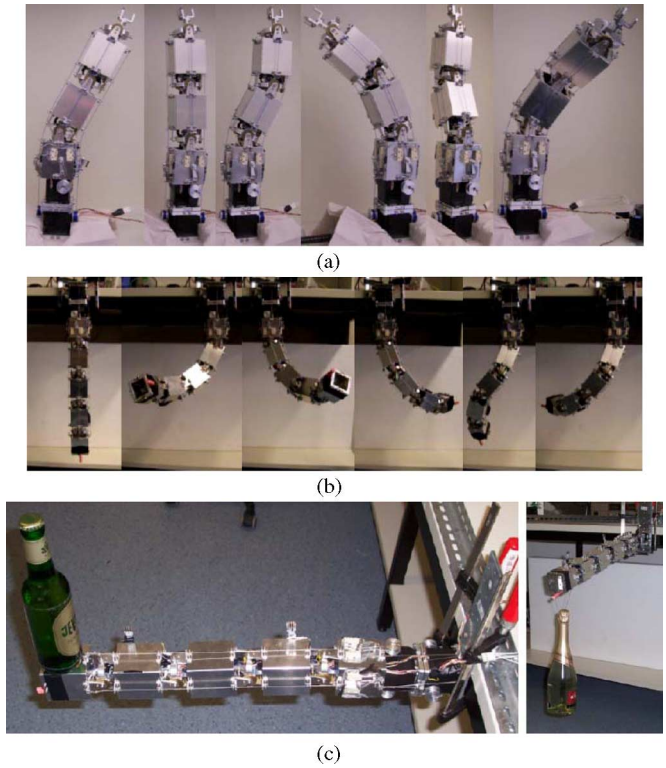


Fig. 11. Snapshot during experiments. (a) 4-DOF. (b) 8-DOF. (c) Test of the locking mechanism of 3D-Trunk (8-DOF) in horizontal pose.

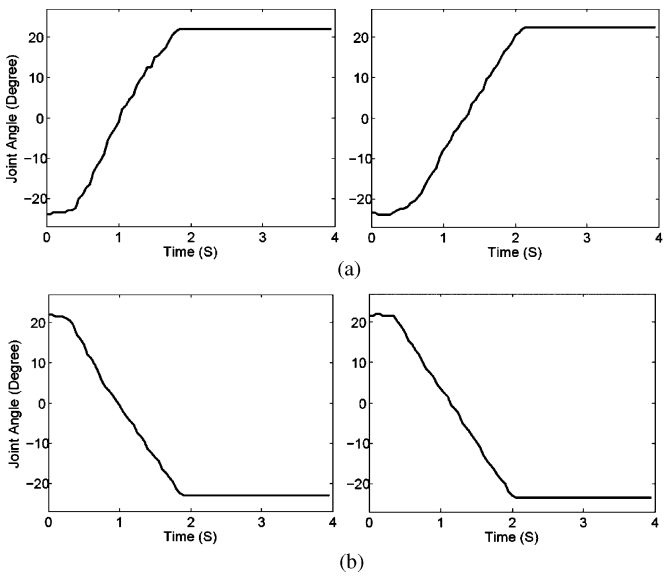


Fig. 12. Two joints' response curves. Sampling frequency is 20 Hz. (a) From -22.5° to 22.5°. Joints 5 and 7. (b) From 22.5° to -22.5°. Joints 5 and 7.

reliable performance and high-load resistance, due to the fact that the available shearing force between "pin" and "hole" is much stronger than that of a friction-based clutch. By contrast, at present, the employed four driving motors are a little weak (slow) for this system. For a future redesign, we need to use stronger motors for getting faster responses and more powerful driving output. The Turn Bracket design shown in Figs. 4 and 5 is too simple, and we get a rather large friction for strongly bent poses. The friction between these Turn Brackets and the Driving-Wires shortens the lifetime of the Driving-Wires and reduces the effective driving power from the motors. Neither of these problems, however, is fundamental, and all can be resolved by small changes in a redesign.

VII. Conclusion and Application Potential

In this paper, we have presented a novel design concept and solution for building a 3-D HRCR. The implemented prototype system "3D-Trunk" is inspired by some ideas from muscle physiology. We use a pulling-only mechanism together with opponent wires for force transfer. This is similar to all muscular-tendon systems, which employ the agonist–antagonist operation to move the joints. Furthermore, we have implemented a novel solenoid-based locking mechanism, which is inspired by the "crystallization-like" process observed in closed clams and mussels, whereby their muscle structure becomes rigid (low energy catch phase [27]) and which eliminates the need for constant holding power. These novel design principles allowed us to build a modular, lightweight system, while still permitting to obtain complex 3-D shapes in a controlled way.

In Section I, we had provided a survey on different HRCRs where many designs differ quite strongly from the one presented here. It is, however, interesting to compare the current approach to that of Ananiev *et al.* [14], who had also utilized a single driving-shared concept but with a solution that is different from the one shown here. They used one irreversible motor to drive a flexible shaft for transporting power, as well as several clutches to distribute selectively the torque and rotation of the flexible shaft. In comparison, our solution is more compact (one CUJC offers 2 DOF) and offers a stronger locking mechanism. The wire-driven mechanism employed by us is more flexible than the shaft-based force transfer in [14].

Our design, however, has also some drawbacks. It cannot perform a "continuous path task" well but has no problems in a "point to point" task. Some other functional properties (e.g., self-motion property) of traditional redundant robots [28], [29] also suffer from the discreteness. At the moment, self-motion can only be observed "on average": When averaging the movements of our HRCR over time, a sequence can be designed such that a given point in space will on average remain stable. Thus, when using a finer discretization in a redesign, the self-motion property will more and more re-emerge, which will be an interesting topic for ongoing research on our design structure.

On the other hand, our design also offers some clear advantages. Because joints are passive and the locking mechanism is very strong, the whole system is lighter than others with the same DOFs. Just several small actuators are required to drive the whole system. The locking resolution is discrete, but the system can still produce many predefined configurations and do so with the high accuracy of a discrete system.

Thus, this system seems especially well suited to be used as shape-changeable, rigid manipulators or positioners due to the fact it can own many highly accurate, predefined configurations. The controllable, binary-state locking mechanism design presented here is strong enough to produce high-locking torques to counteract not only the static loads due the robot's own weight but the dynamic loads produced by the robot's spatial motions as well. This would also make it possible to combine (sequentially connect) traditionally actively driven joints with a shape-changeable system of the kind presented here. As a result, one would get a flexible moveable system with a slower, albeit very strong and precise positioned "limb" built by such an HRCR that can be preshaped into different configurations. Such HRCRs can be designed for 3-D spatial motion or 2-D planar motion. All these applications would require some redesign, but we believe that the principles of the wire-driven shared control together with the novel locking mechanism introduced here may potentially be beneficial and stimulating in the field of hyper-redundant systems.

Acknowledgment

The authors would like to thank the machine workshop of the University of Göttingen for providing them with valuable facilities. The authors would also like to thank the anonymous reviewers for making many valuable comments, which improved the quality of the paper.

References

[1] G. S. Chirikjian and J. W. Burdick, "Kinematically optimal hyper-redundant manipulator configurations," *IEEE Trans. Robot. Autom.*, vol. 11, no. 6, pp. 794–806, Dec. 1995.

[2] G. S. Chirikjian and J. W. Burdick, "A hyper-redundant manipulator," *IEEE Robot. Autom. Mag.*, vol. 1, no. 4, pp. 22–29, Dec. 1994.

[3] G. S. Chirikjian and J. W. Burdick, "Design and experiments with a 30-DOF robot," in *Proc. IEEE Int. Conf. Robot. Autom.*, Atlanta, GA, vol. 3, 1993, pp. 113–119.

[4] E. Shammas, A. Wolf, and H. Choset, "Three degrees-of-freedom joint for spatial hyper-redundant robots," *Mech. Mach. Theory*, vol. 41, pp. 170–190, Feb. 2006.

[5] A. Wolf, H. B. Brown, R. Casciola, A. Costa, M. Schwerin, E. Shamas, and H. Choset, "A mobile hyper-redundant mechanism for search and rescue tasks," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Las Vegas, NV, 2003, pp. 2889–2895.

[6] C. Wright, A. Johnson, A. Peck, Z. McCord, A. Naaktgeboren, P. Gianfortoni, M. Gonzalez-Rivero, R. Hatton, and H. Choset, "Design of a modular snake robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, San Diego, CA, 2007, pp. 2609–2614.

[7] H. Ikeda and N. Takanashi, "Joint assembly movable like a human arm," U.S. Patent 4683406, Jul. 1987.

[8] E. Shammas, A. Wolf, H. B. Brown, Jr., and H. Choset, "New joint design for three-dimensional hyper-redundant robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Las Vegas, NV, 2003, pp. 3594–3599.

[9] G. Long, J. Anderson, and J. Borenstein, "The kinematic design of the OmniPede: A new approach to obstacle traversion," in *Proc. ICRA IEEE Int. Conf. Robot. Autom.*, 2002, pp. 714–719.

[10] G. Granosik and J. Borenstein, "Integrated joint actuator for serpentine robots," *IEEE/ASME Trans. Mechatron.*, vol. 10, no. 5, pp. 473–481, Oct. 2005.

[11] G. Granosik, M. G. Hansen, and J. Borenstein, "The omnitread serpentine robot for industrial inspection and surveillance," *Int. J. Ind. Robots, Spec. Issue Mobile Robots*, vol. IR32–2, pp. 139–148, Mar. 2005.

[12] V. A. Sujan, M. D. Lichter, and S. Dubowsky, "Lightweight hyper-redundant binary elements for planetary exploration robots," in *Proc. IEEE/ASME Int. Conf. Adv. Mechatron.*, Como, 2001, pp. 1273–1278.
[13] K. L. Paap, M. Dehlwisch, and B. Klassen, "GMD-Snake: A semi-autonomous snake-like robot," presented at the 3rd Int. Symp. Distrib. Autonomous Robot. Syst., Saitama, Japan, 1996.
[14] A. Ananiev, I. Kalaykov, E. Petrov, and B. Hadjiyski, "Single-motor driven construction of hyper-redundant robot," in *Proc. IEEE/ASME Int. Conf. Mechatron. Robot.*, Aachen, Germany, 2004, pp. 549–553.
[15] I. D. Walker and M. W. Hannan, "A novel 'elephant's trunk' robot," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatron.*, Atlanta, GA, 1999, pp. 410–415.
[16] H. Ohno and S. Hirose, "Study on slime robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2003, vol. 3, pp. 2218–2223.
[17] L. Roos, F. Guenter, A. Guignard, and A. G. Billard, "Design of a biomimetic spine for the humanoid robot robota," in *Proc. IEEE/RAS-EMBS Int. Conf. Biomed. Robot. Biomechatron.*, Pisa, Italy, 2006, pp. 329–334.
[18] J. Conradt and P. Varshavskaya, "Distributed central pattern generator control for a serpentine robot," in *Proc. Int. Conf. Artif. Neural Netw.*, Istanbul, Turkey, 2003, pp. 338–341.
[19] H. Yamada, S. Chigisaki, M. Mori, K. Takita, K. Ogami, and S. Hirose, "Development of amphibious snake-like robot ACM-R5," in *Proc. 36th Int. Symp. Robot.*, 2005, pp. 433–440.
[20] K. J. Ning, M. Y. Zhao, and J. Liu, "A new wire-driven three degree-of-freedom parallel manipulator," *ASME Trans. J. Manuf. Sci. Eng.*, vol. 128, pp. 816–819, Aug. 2006.
[21] K. J. Ning, "Research and develop of a new type wire driven parallel robot system," M.S. thesis, Northeastern Univ., Shenyang, China, 2002.
[22] S. Kawamura, W. Choe, S. Tanaka, and S. R. Pandian, "Development of an ultrahigh speed robot FALCON using wire drive system," in *Proc. IEEE Int. Conf. Robot. Autom.*, Piscataway, NJ, 1995, pp. 215–220.
[23] K. Maeda, S. Tadokoro, T. Takamori, M. Hiller, and R. Verhoeven, "On design of a redundant wire-driven parallel robot WARP manipulator," in *Proc. IEEE Int. Conf. Robot. Autom.*, Detroit, MI, vol. 2, 1999, pp. 895–900.
[24] J. S. Albus, R. Bostelman, and N. G. Dagalakis, "The NIST ROBOCRANE," *J. Robot. Syst.*, vol. 10, no. 5, pp. 709–724, 1993.
[25] T. Maier and C. Woernle, "Flatness-based control of underconstrained cable suspension manipulator," in *Proc. DETC ASME Design Eng. Tech. Conf.*, Las Vegas, NV, 1999, pp. 1–10.
[26] P. I. Corke, "A robotics toolbox for MATLAB," *IEEE Robot. Autom. Mag.*, vol. 3, no. 1, pp. 24–32, Mar. 1996.
[27] F. Baguet and J. M. Gillis, "Energy cost of tonic contraction in a lamelli-branch catch muscle," *J. Physiol.*, vol. 198, pp. 127–143, 1968.
[28] H. Hanafusa, T. Yoshikawa, and Y. Nakamura, "Analysis and control of articulated robot arms with redundancy," in *Proc. 8th IFAC*, 1981, vol. 14, pp. 78–83.
[29] D. E. Whitney, "The mathematics of coordinated control of prosthetic arms and manipulators," *ASME Trans. J. Dyn. Syst., Meas., Control*, vol. 94, pp. 303–309, 1972.

KeJun Ning received the B.S. (Hons.) and M.S. (Hons.) degrees in mechanical engineering from the Northeastern University, Shenyang, China, in 1999 and 2002, respectively, and the Ph.D. degree in mechatronics engineering from Shanghai Jiao Tong University, Shanghai, China, in 2006.

From May 2006 to July 2007, he was a Senior Mechatronic System Engineer with Shanghai Grandar Robotics Co. Ltd. From August 2007 to May 2008, he was a Senior Engineer and Researcher with the Robotics Department, China Corporate Research Center (CNCRC), Shanghai Branch, ABB (China) Ltd., China. He is currently a Postdoctoral Researcher with the Bernstein Center for Computational Neuroscience, University of Göttingen, Göttingen, Germany. His research interests include the areas of robotics, mechanism design, embedded system, and autonomous systems.

Florentin Wörgötter received the Ph.D. degree in the visual cortex from the University of Essen, Essen, Germany, in 1988.

From 1988 to 1990, he was engaged in computational issues with the California Institute of Technology, Pasadena. He was a Researcher with the University of Bochum, Bochum, Germany, in 1990, where he was engaged in experimental and computational neuroscience of the visual system. From 2000 to 2005, he was a Professor of computational neuroscience with the Psychology Department, University of Stirling, Stirling, U.K., where his interests strongly turned towards "Learning in Neurons." He studied biology and mathematics with the University of Düsseldorf, Düsseldorf, Germany. Since July 2005, he has been the Head of the Bernstein Center for Computational Neuroscience, University of Göttingen, Göttingen, Germany. His current research interests include information processing in closed-loop perception–action systems, sensory processing, motor control, and learning/plasticity, which are tested in different robotic implementations. His group has also developed the RunBot, which a fast and adaptive biped-walking robot.