Distributed Control for an Anthropomimetic Robot

Michael Jäntsch, Steffen Wittmeier and Alois Knoll

Abstract— **Major progress in robotics turns today's humanoid robots into ever safer, more robust, and more agile agents by the moment. However, it is still a long way until robots can safely operate in open environments. Especially in the area of service robotics, the need arises for robots to work flexibly in a human centered environment. One way towards this goal is to incorporate more and more of the mechanisms that can be found in humans for our robots. In this work we would like to propose a bio-inspired control architecture for an equally bio-inspired — namely anthropomimetic — humanoid robot. To achieve this, the human motor control system is analyzed and copied at a structural level. This results in a distributed control infrastructure that is capable of reducing the complexity of the control task by off-loading parts of the control problem into the robot's limbs. Finally, we will prove the fact that it is possible to control an anthropomimetic robot with a large number of degrees of freedom with the proposed control architecture.**

Keywords— **anthropomimetic robot, robot control, distributed control, biomechanics, biorobotics**

Fig. 1. ECCE, a so called anthropomimetic robot, is the platform for the work presented here. In this type of robot not only the outside of the human body is copied but also the inner structures, like bones, joints, muscles, and tendons.

I. INTRODUCTION

Standard humanoid robots mimic the human form, but the mechanisms used in such robots are very different from those in humans. This results in seemingly very unnatural movements even though a very big effort is made in making the trajectories of the robot limbs as smooth as possible. Typically, assemblies of accurately manufactured components are precisely controlled with impressive results, as illustrated by the well-known humanoid robots that have been developed by major Japanese companies, e.g. Honda's Asimo, and Sony's Qrio. However, these robots are still far from matching the abilities of humans in open environments. Additionally, state of the art humanoid robots inherently have severe limitations in their interaction possibilities with humans as the joints are stiff and do not yield to pressure from outside as a human body would. It is true that great advances in robot control [1] have shown that it is possible to build safe robots using standard actuation methods. Nevertheless, the compliance is here mainly added by control, it is not intrinsic to the robot body and comes with the cost of significant computational effort.

For those reasons we believe that the goal of a safe and flexible humanoid service robot comes with the need to not only copy the outside shape of the human body but also its inner structures and mechanisms. Robots that incorporate the

Michael Jäntsch, Steffen Wittmeier and Alois Knoll are with the chair for Robotics and Embedded Systems, Department of Informatics, Technische Universität München, Munich, Germany
`michael.jaentsch@in.tum.de`
`steffen.wittmeier@in.tum.de`
`knoll@in.tum.de`

same mechanisms as humans will also be able to utilize the same tools and operate in the same environments, without posing a larger threat, since they will have similar weight, size, force and dynamic properties. This includes not only a skeleton that is as close to the one of a human as possible, but also compliant muscles. This radically different approach which is called anthropomimetic design [2], leads to more biologically realistic movements. While the technology for building this type of robot has already been explored, it is currently not possible with standard control methods to achieve good control, even though we know that it is possible, because the human brain achieves exactly that. The distinctly human-like movements of an anthropomimetic robot can be attributed to the fact that all movements as well as disturbances are transmitted through the whole robot body by the underlying muscle and joint structure. Simple movements such as lifting an arm will require the actuation of various muscles to retain the body posture. This has to be taken into account, when designing the overall system.

The anthropomimetic approach has already been described by Holland and Knight [2]. Here, the focus clearly lies on the mechanics of the robot and neither the electronics nor the information processing has been taken into account. While this is still at an early stage, first steps towards controlling this complex robot have been made in [3]. Another musculoskeletal robot is Kotaro [4] and its successor Kojiro [5]. Those two robots show an amazing degree of complexity with 91 and 82 degrees of freedom (DoF), respectively, while they are also built to use human-like mechanisms. However, no feasible control strategy has been offered, yet. Possible control strategies for musculoskeletal robots have
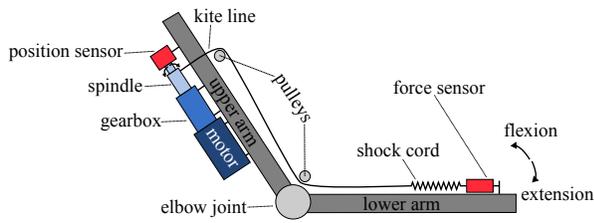
Fig. 2. The actuation principle in an anthropomimetic robot tries to mimic the elasticity of a human muscle and consists of a motor that winds kite line on a spindle and hence exerts a force on the robot's bone.
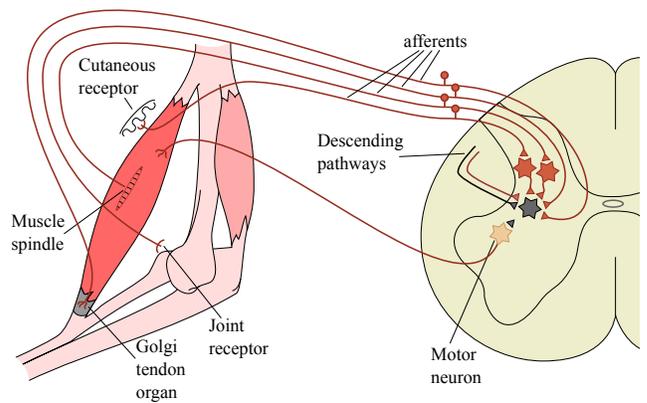

Fig. 3. A motor nucleus is a cluster of motor neurons in the human spinal cord (or brain stem). It is responsible for muscle control and is directly connected to the related receptors [9].

been proposed e.g. by De Sapio et. al. [6] and Kino et. al. [7]. De Sapio et. al. proposed an operational space control scheme for a human shoulder joint, which is easily one of the most complex joints in the human body and Kino et. al. studies the effect of bi-articular muscles on internal forces, while proposing an impedance based control scheme. Before implementing a whole-body control scheme, a control infrastructure capable of supporting it must be present. A distributed control approach similar to the one proposed in this paper, was already mentioned by Blank et. al. [8]. In the latter work, a control architecture using distributed DSPs and a CAN bus for communications is used to control a monopod during highly dynamic applications. However, all results presented were obtained in simulation, while real-world impedance control was only tested for a single degree of freedom.

In this paper we propose the electronic infrastructure for controlling a robot with many DoF and a complex actuator set up, like our anthropomimetic robot, during movement, interaction, and mobile manipulation. This novel control architecture reduces the complexity of the control task by distributing subtasks into the limbs, as it is also done by us humans. In Section II a general overview of the anthropomimetic robot setup is given, while the architecture for controlling this robot is described in Section III, which lays the foundation for future work on controlling this type of robot. Furthermore, results of a first implementation with a simplified test rig featuring a reduced number of DoF, and a benchmark for more DoF are presented in Section IV. Conclusions and Future work are covered in Section V.

II. THE ECCEROBOT

The first anthropomimetic robot CRONOS [2] (see Fig. 1), whose technology is being used in this work, is a robot which tries to mimic the human skeleton, as well as the actuation methods. While the robot bones were made by hand from a thermoplastic which can be hand molded at a temperature of already $60°C$, the artificial muscles (AM) consist of a DC motor, kite line, and shock chord. In this type of electric actuator the motor winds the kite line on the attached spindle and hence either innervates or relaxes the AM, depending on the direction of motor rotation. Therefore, force can only be exerted on the attachment points in one direction — a muscle can only pull, not push. The shock chord adds the flexibility that is also present in a biological muscle (see Fig. 2).

Not only the type of actuation is as close to its biological counter part as possible, but also the attachment points of the AM. Of course it is (currently) impossible to duplicate all of the well over 250 muscles [9] in the human body. To keep complexity at a tolerable level for the first experiments, our current prototype of an upper body has 45 AM, but in due time we are planning to build another prototype with ~ 80 AM. The muscles that were chosen to be duplicated in the current prototype are the ones responsible for larger scale movements, omitting the ones used for fine grained dexterous movements, e.g. in the hands (for now).

In all robots, as well as humans, proprioception — the sense of the relative position of neighboring parts of the body — is fundamental for well-controlled movements and interactions with the environment. The use of high-precision actuators in conventional robots allows for the direct measurement of the joint positions which can easily be mapped to a body pose. In an anthropomimetic robot, however, the use of compliant actuators makes measuring the motor positions insufficient, since the lengthening of the shock chord cannot be determined. Generally it is also problematic and error prone to measure the joint angles directly, because most joints in the human body are no simple hinge joints with a single DoF, but so called ball-and-socket joints with 3 DoF.

We believe that good proprioception for this highly bio-inspired multi DoF robot cannot be achieved by traditional methods, and therefore the the sensory system, as well, needs to be bio-inspired. A human muscle has two types of embedded sensors (see Fig. 3). One is the so called muscle spindle, which is a sensory receptor encapsulated in the fleshy part of the muscle. The other is the golgi tendon organ which is located at the attachment point of the tendon to the muscle fibers. While muscle spindles are most sensitive to changes in the muscle length, the tendon organs mainly measure the muscle tension [9]. In the human body there are also joint angle sensors, but due to their inaccuracy they can only deliver very rough estimates of the angles. The cutaneous receptors give a feedback of the muscle tension by measuring the stretch in the skin covering

the muscle. This sensor is redundant to measuring the muscle tension, and can as well achieve only low resolution sensing. To mimic this behavior in the robot, each of the actuators will be equipped with a set of sensors, measuring the motor position, the tendon strain, and the motor current. While the current sensor is used for direct control of the DC motor and is necessary due to the nature of DC motors, the motor position and the force sensor can be used to obtain the data that in a biological muscle would be obtained by the muscle spindles and the tendon organs. the muscle length from the motor position and the force (see Fig. 2). As all joints are spanned by multiple muscles, knowing the lengths of all AM is sufficient to calculate joint positions and therefore achieve proprioception. For this reason and due to the inaccuracy of both the cutaneous sensors and the joint angle sensors, both are not strictly necessary.

III. THE DISTRIBUTED CONTROL ARCHITECTURE

In an anthropomimetic robot, all body loads are transferred throughout the structure due to the elasticity in the AM, which without additional control is highly under-damped. This leads to the fact that, unlike traditional robotics platforms, all limb movements and robot-environment interactions are whole-body movements. The flexibility that is being added to the robot poses huge problems on the design of control algorithms. For this reason it is highly unwanted for conventional robots On the other hand, without it, it might never be possible to achieve human-like motion in an artificial robot. The other problem is the immense number of DoF in the skeleton together with the highly redundant setup of the AM.

Typically, robot control is done using a centralized control scheme, where all sensors and actuators are connected to a single controller. The control algorithm that is being executed on this central controller fetches the sensor values and calculates the actuation for all joints in a single step. This is possible when the number of DoF is limited. For an anthropomimetic robot with approximately 80 AM and 3 sensors per AM this approach becomes infeasible. One reason is the cabling, which would be enormous in the complexity and also the pure weight. The other is the complexity of reading 240 sensors on a single centralized controller.

In the human body on the other hand, motor control is organized in a hierarchy. While most of the low level control takes place in the spinal cord and brain stem, voluntary motor control commands are issued by the fore brain. Typically a muscle is controlled directly by a set of motor neurons in the spinal cord that form a motor nucleus [9]. The fore brain can issue commands to the motor nuclei through descending pathways (see Fig. 3). The existence of the motor nuclei shows that in the human body the control is highly distributed, where fast low level control is conducted as close to the muscles as possible and the higher levels of (voluntary) movement control communicate with the muscles through the distributed units (motor nuclei). Voluntary reaction times range from $60\,\mathrm{ms}$ to $120\,\mathrm{ms}$ and can get as low as $40\,\mathrm{ms}$ for

reflexes [9], which shows that latencies in the human body are actually much higher than in today's robots where control algorithms run at frequencies up to $2\,\mathrm{kHz}$ or higher. Still a human is capable of achieving high-speed motions through feed-forward control, by exploiting the intrinsic dynamics of the body and nervous system.

A robust control architecture is to be designed, which can reduce the complexity of the control task. One way to do this is to stay close to the human archetype and distribute processing units (motor nuclei) around the robot's body to be as close to the sensors and actuators as possible. Each of the boards is connected to a central controller (fore brain) via a communication bus and therefore only a single bus link plus electric power needs to be routed to the boards. This reduces the cabling, since power and information can be distributed in a tree-like manner. In [4], Mizuuchi et. al. propose a control architecture, where sensor data and motor commands are transferred via the Universal Serial Bus (USB) to distributed nodes. In this setup, however, there is no processing in the distributed nodes. The control algorithm itself is still centralized.

As was previously done in [8] for a monopod with only 2 DoF, we propose to implement fast local control loops, like force, position, and impedance control, as well as local reflexes, like the stretch reflex[1], on the distributed nodes (see Fig. 4). Additionally sensor preprocessing and fusion, like the calculation of the muscle length from the force and motor position sensors, will also be executed on the distributed nodes. Each of the AM can be seen as a unit, including the electronics, the motor, the tendon, and the associated proprioceptive sensors. The distributed control nodes are linked to the central controller via a bus system. Although, we decided against the fast USB bus used in [4], in favor of the much slower Controller Area Network (CAN). The reason for this is that CAN is only a two wire serial bus and unlike USB no hubs are required. As the hubs would need to be fitted on the robot body, their weight and size would turn into an additional challenge. Furthermore CAN features the possibility for any bus participant to broadcast messages on the bus, which will turn out to be a useful property for the implementation of fast low-level reflexes.

There have been several hypotheses on how human motor control works, e.g. the so called equilibrium-point hypothesis (EPH) [10] or the internal dynamics model hypothesis (IDMH) [11] and it is widely disputed which one yields the better explanation of human motor control. However, both share the insight that it is not necessary for the generation of high-level motor commands to receive high-frequency sensor data. We are confident that the communication bandwidth can be reduced by distributing the control task in a bio-inspired way. The required bandwidth can be estimated, when assuming the transfer of three sensor values (at $12\,\mathrm{bit}$) and one motor control command (at $12\,\mathrm{bit}$) per AM, at a control frequency of $50\,\mathrm{Hz}$. Additionally an extra $2\,\mathrm{bit}$ for

---

[1]The stretch reflex leads to counter muscle activation in case of sudden muscle lengthening [9].
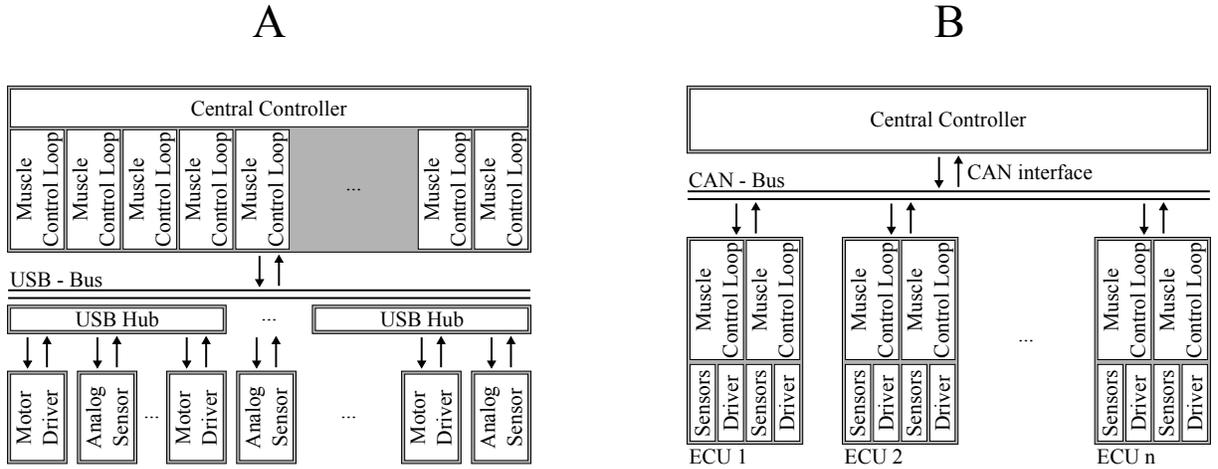
A   B

Fig. 4. In a centralized control architecture (Fig. A) like it is used for Kotaro [4] all control is run on a single controller. The control architecture proposed here (Fig. B) distributes fast local control loops into the robot's limbs, while communication with the central controller is accomplished via a CAN bus system.

the control type is reserved. As mentioned above, human reaction time is even lower than that. Therefore assuming $50\,\mathrm{Hz}$ is sufficient from our standpoint.

$$BDW = 80 \cdot ((2\,\mathrm{bit} + 12\,\mathrm{bit}) + 3 \cdot 12\,\mathrm{bit}) \cdot 5\,\mathrm{Hz}$$
$$\approx 195\,\mathrm{kbit/s} \qquad (1)$$

Even when taking the necessary overhead into account this shows that using CAN for communication is sufficient[2]. To further reduce the bus load and latencies it is also possible to use several buses and therefore split the communication load. In our case of using 80 AM we chose to have two buses, which also reduces the participants and therefore the collisions on the bus to a tolerable number.

For this purpose we developed electronic control units (ECUs) for sensor-actuator control. Each of those ECUs has enough processing power to run the control algorithms and sensor preprocessing for two AM. Preliminary simulations of a single muscle have shown that the local control loops should run at a frequency of $500\,\mathrm{Hz} - 1\,\mathrm{kHz}$ to ensure stability. That is slightly slower than the frequency previously mentioned for the control of standard robots. This can be attributed to the longer time constants of the system. The flexibility that is being added to the actuators also makes the whole system slower. To effectively reduce cabling and avoid running vulnerable analog sensor signals along the robots limbs, the ECUs have to be placed in strategic locations around the robot torso. Therefore size and weight are critical factors. Ideally, each actuator would have a dedicated ECU that fits right behind the motor. In this case, however, the board size would be only slightly smaller than in the case where each ECU incorporates two AM and the number of boards that would have to be fit on the robot body would double. The approach of using an ECU for three or more actuators on the other hand, is also not feasible as the probability of power wires that will need to be routed past joints will increase.

[2]CAN has a maximum bandwidth of $1\,\mathrm{Mbit/s}$.

Fig. 5 depicts a distributed control unit, as it was developed for this specific project. It features an STMicroelectronics STM32F microcontroller, incorporating a 72 MHz ARM Cortex-M3 processor, several Analog-to-Digital converters and an integrated CAN interface, as well as power electronics for two motors. The motors are controlled by PWM, using two full H-Bridges. Direct feedback is given by an integrated hall-effect-based current measurement unit in the motor loop.

The firmware is developed to implement a finite state machine (FSM) which can be easily controlled from the central processing unit, using a custom communication protocol developed especially for this purpose. This protocol is based on raw CAN and defines messages needed to initiate state transitions. While most of the messages are unacknowledged, some require a reply from the distributed node to be able to determine communication or controller failure. At the same time, a heartbeat is broadcasted to all nodes, so that each of them can determine central controller or communication failure on its own. In case a node detects an error it will go into a failure state and stop replying to messages (fail-silent behavior [12]). As each processing unit handles the control of two AM, the FSM will be instantiated twice on each ECU. In the *On* state the local control algorithm will be executed at a fixed frequency, while a number possible control schemes, like force, impedance or position control can be implemented. At a higher level preprocessed sensor values can be used to achieve proprioception.

To be able to handle the complexity of a distributed system with ~ 40 nodes, the nodes can be parameterized dynamically at run-time, while the firmware image is the same for all nodes. Parameters that can be changed dynamically include the control parameters of the different control loops as well as general control parameters. Only a unique identifier and bootloader needs to be stored in flash, once, while the software image can be exchanged easily via the CAN bus during system startup.

Fig. 5. The electronic control unit (ECU) running the distributed controllers, which was specifically developed for this project.

IV. EXPERIMENTS

Several experiments were performed to verify that the proposed infrastructure can be used to control multi DoF robots. First, the bus latency was examined, using a set up with two processing units. Second, experiments were made to show that it is possible to scale the architecture to a full robot with 80 AM, and last, it was shown that the developed control architecture, can be used to control a robot arm (see Fig. 6), featuring the same mechanisms as the full robot but only eleven AM.

A. Examining Bus Latency

The theoretical bus latency for a full CAN frame with a payload[3] of $8\,\mathrm{Byte}$ and therefore the lower bound of the latency that can be achieved when using CAN to transport $8\,\mathrm{Byte}$ can be calculated as:

$$t = \frac{108\,\mathrm{bit}}{1\,\mathrm{Mbit/s}} = 0.103\,\mathrm{ms} \qquad (2)$$

This can be verified by measuring the communication round-trip time between two of the ECUs with an internal hardware timer on one of the μ-Controllers. In the configuration used for this experiment, the timer had a resolution of $10\,\mu\mathrm{s}$. The average round-trip time of the message was determined to be $0.247\,\mathrm{ms}$ (see Table I). This time includes twice the bus latency plus twice the processing time for sending and receiving. As the CAN interface on the μ-Controllers is realized in hardware, the processing time on the processor itself can be neglected in this experiment. Therefore the bus latency (including the hardware processing time for sending and receiving) can be calculated as $\sim 0.124\,\mathrm{ms}$.

B. Scaling the Control Architecture to 80 AM

A control cycle consists of three phases, first fetching sensor values, second calculating new control variables, and third setting the control signals. In the case of the architecture presented in this paper, the fetching of the sensor values as well as the setting of control variables needs to be performed via the bus system. For this experiment a laptop with a current (2009) Intel dual-core processor, running Linux is used. To ensure high priority interrupt handling and process

[3]A CAN message can carry up to 8 Byte of data [13]

TABLE I
THE TIMES ARE ROUND-TRIP LATENCIES ON THE CAN BUS, MEASURED
BY SYSTEM TIMERS (σ IS THE STANDARD DEVIATION).

	Exp.	Max.	Min.	Av.	σ
Single Message	10000	$0.25\,\mathrm{ms}$	$0.24\,\mathrm{ms}$	$0.247\,\mathrm{ms}$	$0.0046\,\mathrm{ms}$
Control Cycle	100	$5.94\,\mathrm{ms}$	$4.90\,\mathrm{ms}$	$5.28\,\mathrm{ms}$	$0.46\,\mathrm{ms}$

scheduling, the preemptive kernel patch by Ingo Molnar[4] is used. The possibility of controlling 80 AM with a frequency of $50\,\mathrm{Hz}$ with only two buses can be verified as follows:

The communication on the two buses can be fully parallelized, so in this experiment only communication with half of the AM is examined. One sensor request message (payload: $0\,\mathrm{Byte}$) is broadcasted to the controllers, subsequently waiting for 40 sensor data messages (payload: $5\,\mathrm{Byte}$). Finally 40 motor control messages (payload: $3\,\mathrm{Byte}$) are sent. The necessary communication time is measured using the internal clock of the PC and amounts to an average latency of $5.28\,\mathrm{ms}$ (see table I). This shows that it will be possible to easily run whole-body control with a frequency of $50\,\mathrm{Hz}$ with the given set up. The resulting period of $20\,\mathrm{ms}$, leaves at least $14\,\mathrm{ms}$ for the computation of the control algorithm itself.

C. Controlling an Anthropomimetic Robot Arm

An anthropomimetic robot arm (see Fig. 6) with 11 AM was used to verify the architectural design. This robot arm uses the same mechanisms as the full robot, and therefore results are expected to apply to the full robot as well. However, due to the reduced number of DoF, the challenges in implementation are reduced.

All AM were simultaneously controlled with the distributed control approach, while control was performed using the motor position sensors only. New motor positions, mapping to a body pose, were handed to the local control loops. Under this scheme, simple movements like shoulder and elbow abduction and adduction, and shoulder anteversion and rotation were performed. Even though all AM were directed simultaneously to the goal position and no specific trajectory control was performed, the movements that were achieved by this control scheme were strikingly smooth and seemed to the subjective observer highly human-like. Already simple control schemes lead to a behavior that seems a lot more human-like than in traditional humanoid robots, because a lot of the control effort was off-loaded into the robot's body. While on the one hand the control task is performed by the distributed control units, the flexible bio-inspired muscle setup further reduces the need for exact time synchronization, because communication between the muscles is off-loaded into the morphology. By copying the human motor and sensor system as well as the distributed control architecture,

[4]This patch makes kernel preemption possible for almost all areas in the kernel, and therefore reducing the latency for high-priority tasks. At the same time it deals with priority inversion by exchanging all kernel spin-locks with an implementation using priority inheritance.
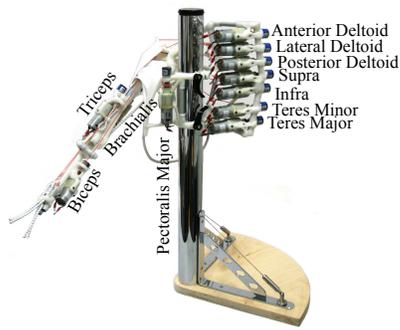
Fig. 6. The anthropomimetic robot arm with 11 AM that was successfully controlled by the proposed control infrastructure.

morphological computation, as mentioned by Pfeifer et. al. [14], is automatically exploited in a way similar to the human body.

V. Conclusions and Future Works

A. Conclusions

In this paper a novel, distributed control architecture for compliant robots with many DoF was presented. The anthropomimetic design and the presence of compliance presents unique challenges in the design of the control system. The solution proposed is compatible with the highly bio-inspired principles with which the robot body was designed and constructed. State of the art control infrastructures do not feature the necessary performance in a set up of this type. The human-like skeleton and flexible muscles, require a control infrastructure that is inspired by the human neural system, featuring distributed motor nuclei and a centralized controller (fore brain) to issue voluntary movement control. While the exact control scheme is left open, the architecture is kept as flexible as possible to allow for the implementation of different possibilities.

The approach greatly reduces the complexity of the control task, by off-loading tasks into the robot's body. We were able to show with several experiments determining system latencies that it will be possible to control a robot like the anthropomimetic robot described in section II and that the proposed infrastructure will also scale well to a robot with 80 AM. Furthermore an implementation of the proposed control architecture was used to control an anthropomimetic robot arm, copying a human shoulder and elbow joint. It is noteworthy that even a simple control scheme produced very smooth trajectories and strikingly human-like movements.

B. Future Work

In the near future we plan to implement and refine different robot control schemes within the proposed infrastructure. The goal is to be able to control the robot during dedicated movements like interaction with the environment and object manipulation. It will be necessary to add further sensors, namely vision, inertial and touch sensors and utilize the additional information along with the existing proprioceptive sensors. Vision is particularly important as proprioception

alone will not be accurate enough to do fine grained dexterous tasks with a compliant robot. In the human body the extraordinary vision system accounts for the fact that the model used for controlling the limbs is very inaccurate and underlies frequent changes. Therefore vision can be incorporated for feedback control (visual servoeing) of e.g. the hand when reaching for objects, etc.

Furthermore, research will be conducted on verifying that the trajectories observed in the anthropomimetic robot arm, which seemed to the subjective observer highly human-like, truly resemble the motions observed in the human body. By doing this we hope to find further evidence on the exploitation of morphological computation in an anthropomimetic robot. Hopefully this project will give us additional insights about human motor generation, while exploiting the anthropomimetic nature of the robot to achieve some human-like cognitive characteristics.

VI. Acknowledgments

References

[1] S. Haddadin, A. Albu-Schaffer, A. De Luca, and G. Hirzinger, "Collision detection and reaction: A contribution to safe physical human-robot interaction," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems IROS 2008*, 2008, pp. 3356–3363.

[2] O. Holland and R. Knight, "The anthropomimetic principle," in *Adaptation in Artificial and Biological Systems*, 2006.

[3] O. Holland, H. G. Marques, and R. Newcombe, *Controlling an Anthropomimetic Robot: A Preliminary Investigation*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2007, vol. Volume 4648/2007, pp. 736–745.

[4] I. Mizuuchi, T. Yoshikai, Y. Sodeyama, Y. Nakanishi, A. Miyadera, T. Yamamoto, T. Niemela, M. Hayashi, J. Urata, Y. Namiki, T. Nishino, and M. Inaba, "Development of musculoskeletal humanoid kotaro," in *Proc. IEEE International Conference on Robotics and Automation ICRA 2006*, 2006, pp. 82–87.

[5] I. Mizuuchi, T. Nakanishi, Y. Sodeyama, Y. Namiki, T. Nishino, N. Muramatsu, J. Urata, K. Hongo, T. Yoshikai, and M. Inaba, "An advanced musculoskeletal humanoid kojiro," in *Proc. 7th IEEE-RAS International Conference on Humanoid Robots*, 2007, pp. 294–299.

[6] V. De Sapio, J. Warren, and O. Khatib, *Predicting reaching postures using a kinematically constrained shoulder model*. Springer Netherlands, 2006, ch. 3, pp. 209–218.

[7] H. Kino, S. Kikuchi, T. Yahiro, and K. Tahara, "Basic study of biarticular muscle's effect on muscular internal force control based on physiological hypotheses," in *Proc. IEEE International Conference on Robotics and Automation ICRA '09*, 2009, pp. 4195–4200.

[8] S. Blank, T. Wahl, T. Luksch, and K. Berns, "Biologically inspired compliant control of a monopod designed for highly dynamic applications," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems IROS 2009*, 2009, pp. 148–153.

[9] E. R. Kandel, J. H. Schwartz, and T. M. Jessel, *Principles of Neural Science*, 4th ed., J. Butler and H. Lebowitz, Eds. McGraw-Hill, 2000.

[10] A. G. Feld'man, "On the functional tuning of the nervous system in movement control or preservation of stationary pose. II. Adjustable parameters in muscles," *Biofizika*, vol. 11, no. 3, pp. 498–508, 1966.

[11] M. Kawato, "Internal models for motor control and trajectory planning." *Curr Opin Neurobiol*, vol. 9, no. 6, pp. 718–727, Dec 1999.

[12] J. Reisinger and A. Steininger, "The design of a fail-silent processing node for the predictable hard real-time system mars," *Distributed Systems Engineering*, vol. 1, no. 2, pp. 104–111, 1993. [Online]. Available: http://stacks.iop.org/0967-1846/1/104

[13] *Road vehicles – Interchange of digital information – Controller area network (CAN) for high speed information*, ISO 11898 Std., 1993.

[14] R. Pfeifer, F. Iidaa, and G. Gmeza, "Morphological computation for adaptive behavior and cognition," in *International Congress Series*, vol. 1291, 2006, pp. 22–29.