Myorobotics

A Framework for Musculoskeletal Robot Development

Background

Compliant, musculoskeletal robotic systems offer several advantages, especially in situations where human and robot work in close proximity. A musculoskeletal design takes inspiration from the mechanics of the human body. It makes extensive use of viscous-elastic materials to emulate the muscles and tendons which enhance safety, dexterity and adaptivity in uncertain environments. It also allows reducing body weight and developmental cost, while at the same time increasing design flexibility.

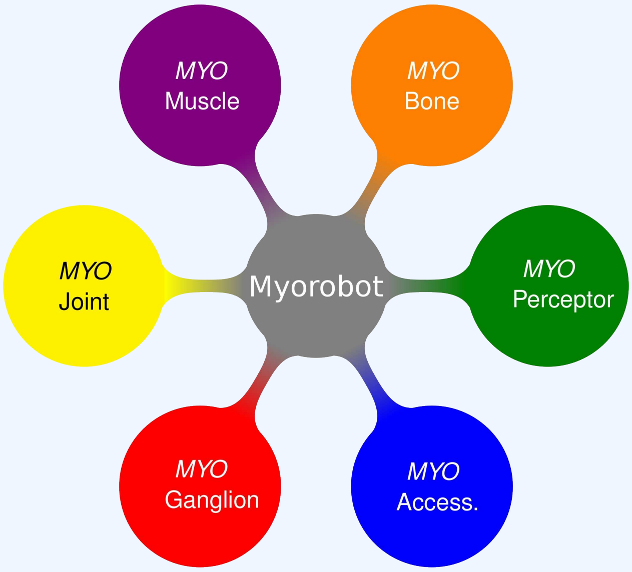
Although there are several research platforms available that employ this design, current systems utilize custom-made, complex hardware and software, which inhibits their use beyond robotics research in academic settings.

Project Goal

The goal of this project is to develop the ***Myorobotics toolkit***, a commercial, modular and reconfigurable system for developing musculoskeletal robotic platforms. It is designed for use by experimenters of different disciplines and aims to allow them to create, configure and operate their experimental setups based on their individual needs. In addition to academic settings, the toolkit also targets the industrial sector for applications that require the capability to mimic biological structures while maintaining high flexibility and reasonable costs.

Our approach

What we would like to achieve is to maintain as much as possible of the capability to design and build engineered bio-inspired robots, while offering the modularity and flexibility offered by construction toolkits. For this reason, we will use engineered modules which mimic the functions of their biological equivalents, e.g. muscles, bones, joints, etc. These modules, which we call ***design primitives***, can be configured and then assembled to custom-made robots.



Winner of the 2013 German High-Tech Champions Award (http://is.gd/ghtc2013)

Use of integrated elecromechanical interfaces reduces cable clutter and simplifies assembly. We have also decided on a two generation model, where feedback from actual use of the first generation of design primitives will lead in improvements and revisions for the second one.

In parallel to the development of the two generations of design primitives, we will be investigating the most suitable materials and production techniques to reduce cost, but also considering practicality, availability and performance.

In addition to the hardware, the Myorobotics toolkit includes a controller library and a suite of software tools that cover the major use cases and facilitate the development of user-driven extensions. The controller library provides low level controllers and the APIs needed to facilitate the development of high level controllers by end-users. Nevertheless, two high level controllers are also provided for demonstrating the toolkit's capabilities.

The software tools, bundled together as the Myorobotics Development Environment (MYODE), provide the features necessary for designing, configuring, simulating, operating and optimizing robotic assemblies within an easy to use, graphical user interface. Additionally, MYODE includes importing and exporting facilities that enable the development of new design primitives and the in-house production of the components necessary for a Myorobotics assembly.

Since our goal is to make this toolkit usable even by non-experts, we are also developing a self-diagnosis and a self-calibration system that will be part of MYODE. These systems will reduce the effort required to maintain a system and improve its reliability.

Design primitives

Myorobotics design primitives represent a new class of modules for professional robotics. In contrast to conventional robots, the design primitives are strongly inspired from the human and animal musculoskeletal system, allowing the user to create Myorobots mimicking biological limbs. Each type of design primitive therefore corresponds to a biological entity achieving a musculoskeletal, proprioceptive or exteroceptive function. We have defined six types of design primitives (see Figure on the left); bones, joints, muscles, ganglions, perceptors and accessories, the latter being supporting modules such as the power supply. Recently the 1st generation was completed and is present in the following sections.

Project Summary - September 2013

For the latest information visit us at: http://myorobotics.eu/

Bones

MYO-Bones are passive mechanical structural modules and the basic building-blocks forming the kinematic chain, or skeleton, of any Myorobot. Their main characteristics are lightweight design and high stiffness, while providing fixation points for the other design primitives.

MYO-Bones are currently implemented using parallel CFC tubes bundled by transversal aluminum spacers. Leveraging techniques such as water-jet cutting, the spacers are easy to produce, while the size of the tubes can be adjusted to create MYO-Bones with various lengths. Thanks to its open structure, this implementation allows to integrate easily the electric cabling in the MYO-Bone structure and offers broad fixation possibilities. The ends of the MYO-Bone feature a new force-transmitting, electromechanical quick-connecting element to mount MYO-Joints.

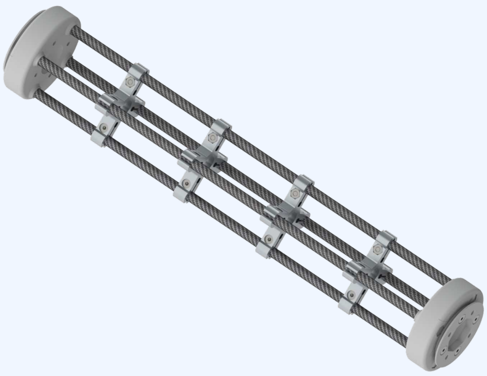
Joints

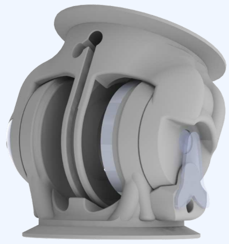
MYO-Joints are passive mechanical connector modules, complementing the MYO-Bone to form the skeleton of any Myorobot. Several types of MYO-Joints are envisioned, covering the diversity of articulations found in nature. MYO-Joints production strongly leverages additive production techniques, such as selective laser sintering (SLS) or fused deposition modeling (FDM), allowing cost effective production, optimized mechanical design and compact integration of the sensors and electric cabling.

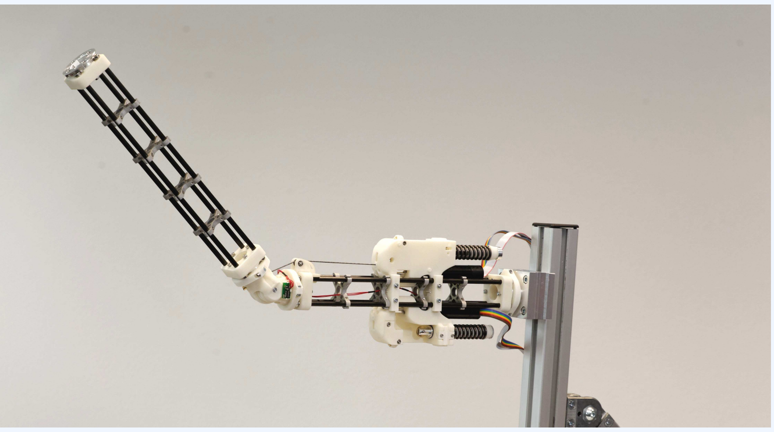
So far, two kinds of MYO-Joints have been developed, mimicking hinge and pivot joints. Both are using ball bearings for low friction operation and are equipped with absolute angular sensors.

Muscles

MYO-Muscles are modules mimicking the function of skeletal muscles. For this reason, MYO-Muscles differ from conventional robotic actuators in several ways. The most important one is that MYO-Muscles are intrinsically compliant actuators, whereas conventional robotic actuators are essentially rigid. Compliance is implemented with an adjustable elastic element arranged in series with the muscle contractile element (typically an electric DC motor). This plays a crucial role in reducing the influence of external perturbation of the motion (easier control), reducing the impact force during unexpected collision with the environment (improved safety) and enabling elastic energy storage in the muscles to achieve highly dynamic motions (improved performance). Additionally, MYO-Muscles are, as muscles, a unidirectional actuator (i.e. they only generate force under tension, not compression) connected to the skeleton using cables, mimicking the function of the biological tendons. Bidirectional actuation of a joint therefore requires two antagonist muscles, while one muscle can actuate one or two joints (resp. monoarticular and biarticular muscles).





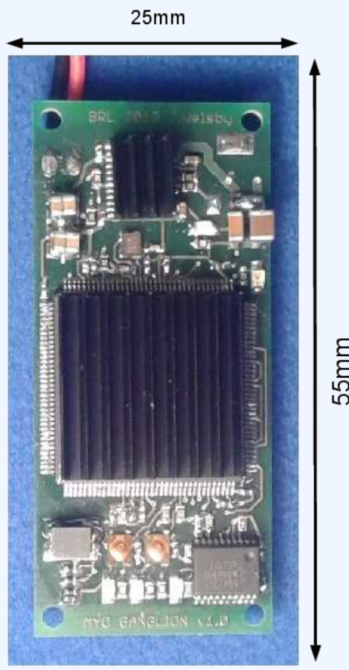


Myorobotics 1-DoF assembly consisting of two bones, a joint and two muscles

The current MYO-Muscle implementation is based on a geared DC motor (combined with a winch to wind up the tendon cable) and a compression spring as the adjustable elastic element. These are integrated, together with a sensor measuring the deflection of the spring (hence the muscle generated force) and several guides and pulleys to redirect the tendon cable, in a cost-efficient plastic housing produced with SLS or FDM. MYO-Muscle design allows the compact integration into the structure of the MYO-Bones, where a maximum of four MYO-Muscles can be mounted using special fastener elements similar to the spacers.

Ganglions

MYO-Ganglions are high speed communication and data processing units, able to collect sensory information from other Design Primitives, control one or more MYO-Muscles and communicate with other MYO-Ganglia. Communications between MYO-Ganglia and with the external control system, which require high speed and bandwidth, are based on the automotive network communications protocol FlexRay (up to 10Mbit/s), while communication with local sensors relies on the highly flexible but slower CAN Bus. Like their natural pendant, MYO-Ganglia also allow the execution of vegetative (i.e. autonomous) control strategies.

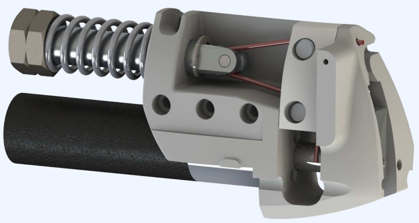


A MYO-Ganglion; the heatsinks cover the microcontroller and the voltage regulator

Perceptors

An additional type of design primitives envisioned for the future development of the toolkit are MYO-Perceptors. These are sensor modules that will provide exteroceptive sensory functions, such as vision, tactile sensing and so on, ultimately needed to build and operate Myorobots that are aware of their environment.

Renderings of different design primitives. From left to right: bone with structural bonds on either side, hinge joint, pivot joint and muscle with integrated elasticity.



Controller Library

The controller library will provide a minimum set of controllers to be used with Myorobots as well as the basis for the development of new controllers by end-users. At the lowest level, we have already implemented linear feedback controllers for the muscles. These controllers are implemented on the MYO-Ganglion as a PID controller, with an additional feed-forward term using the desired set-point (sp), and optional dead-band and integral wind-up limiting.

Additionally, two types of high-level controllers are currently under development. The first is based on cerebellum-inspired control and in particular on the motor control of mammals that will be implemented on single- and multi-joint assemblies. The second type refers to a class of controllers which allows to form meaningful reflex circuits in different modalities and then exploit these reflexes to achieve coordinated behavior. The second

Myorobotics Development Environment

Myorobotics provides a software toolchain called Myorobotics Development Environment (MYODE) which facilitates design, production, testing and operation of musculoskeletal robots. It is designed as an easy to install and use graphical environment. The MYODE tools include among others the robot abstraction, the virtual assembly and the optimization tools, that are currently in an advanced stage of development. Developers will find a comprehensive API documentation, an easy to use build system and a repository for non-standard dependencies that are kept to a minimum.

MYODE is implemented as a set of plug-ins for Caliper, an extensible robotics development environment for tendon-driven robots that is already in the final stages before release. Caliper comes with the following basic functionality:

- ◇ Plug-in management
- ◇ Central logging facility (usable by all plugins)
- ◇ Settings manager to facilitate setting and storing settings relevant to Caliper or individual plugins
- ◇ Configurable, persistent workspace.
- ◇ Simulation, data acquisition and terminal plug-ins.

Simulation plug-in

The main features of the simulation plug-in are:

- ◇ A backend supported by the Bullet physics engine.
- ◇ An extension mechanism using Bullet callback mechanisms that allow users to create customized actuators and constraints in the physics engine.
- ◇ Capacity to run multiple simulator instances.

Data acquisition plug-in

The data acquisition plug-in facilitates the collection of data and their visualization. All plug-ins can register their own data sources which are then managed through a single, easy to use graphical interface. Support for multiple instances is also provided, making it easier for users to monitor a large number of data sources simultaneously.

Terminal plug-in

The terminal plug-in provides a command-line interface to other plug-ins. Specifically, a plug-in registers the available commands along with the arguments they take and how they map to plug-in interfaces. To the user each connected plug-in is represented as a "virtual folder". User input is facilitated with a comprehensive, context-sensitive auto-completion facility.

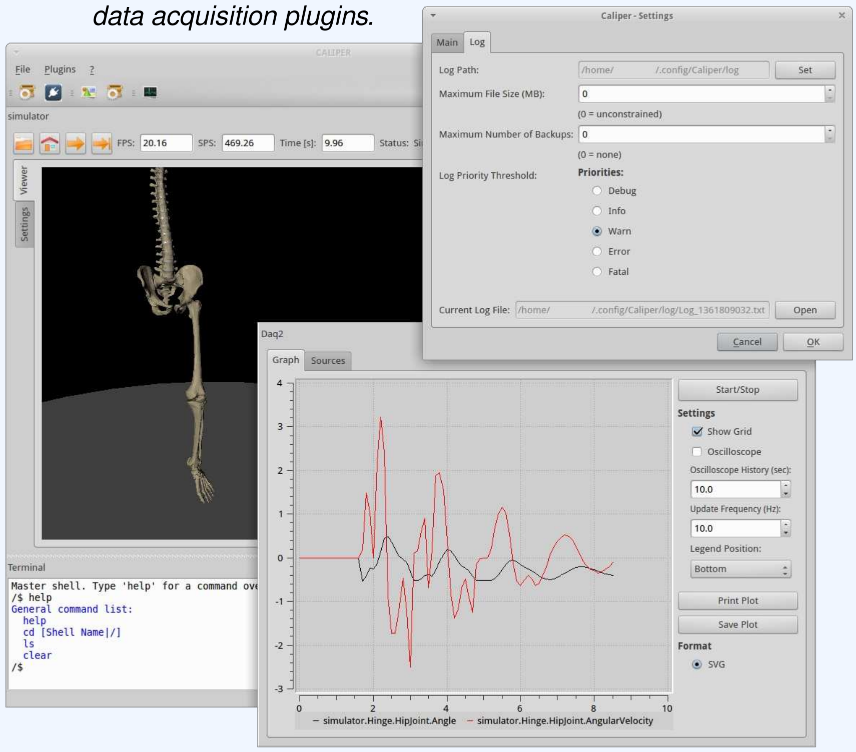
Virtual Assembly plug-in

This plug-in allows users to create, edit and experiment with different assemblies in an intuitive way. An assembly is defined as "two or more design primitives assembled together" and can represent anything up to very complex robots. The user interface consists of multiple elements that include the assembly editor visualized as a block diagram, a 3D preview, a property editor and a design primitive library viewer where all available design primitives are listed.

Robot Abstraction plug-in

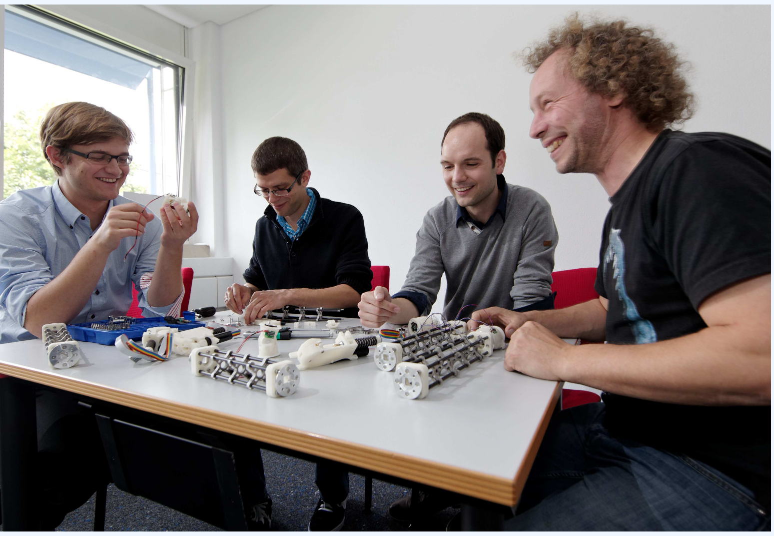
The robot abstraction plug-in provides an interface to send commands to and receive sensor information from a physical robot or a virtual robot inside a simulator instance. The use of a common interface facilitates the transparent switching between operating simulated and physical systems that is useful in several scenarios, e.g. controller development.

Screenshot of Caliper showing the simulator, terminal and data acquisition plugins.



Controller Optimization plug-in

The controller optimization tool allows a user to optimize a given controller on their designs in a given task. For the optimization, the tool uses an evolutionary algorithm which mimics the process of natural evolution. The algorithm optimizes a population of individuals overall several generations. In each generation the best performing individuals are selected and modified to create the population for the next generation. The user needs only to specify the list of parameters to be optimized and a fitness function (e.g. energy efficiency), according to which the population individuals are scored and selected.



From the meeting where the 1st generation of design primitives was delivered and demonstrated.

Self-Diagnosis plug-in

This tool extracts sensor and motor correlations from a working system, and then uses them to assess whether the system has changed or not. It works in two stages: the identification stage and the diagnosis phase. The first stage consists of corellating all the sensor and motor signals induced by fixed feed-forward commands. In the second stage a new correlation matrix is identified and compared with that obtained in the first run. Elements in the matrix that differ substantially indicate possible faults.

Expected outcome & impact

Towards the end of the project we will release the Myorobotics toolkit with an open-source, free to use for non-commercial purposes license. By that time, we plan to have a complete set of design primitives that includes more bone and joint modules of different sizes, improved muscle units and exteroceptive sensors. The MYODE toolkit will also be complete and user-tested, while the controller library will be finalized. The end of the project will also coincide with a comprehensive demonstration of the toolkit featuring all the steps from assembly to operation of different robots.

In the mid- to long-term we expect our toolkit to lay the foundations for engineering a number of musculoskeletal systems based on different application requirements. The use of hardware primitives will improve component reuse and lower development costs. MYODE will facilitate the design, operation and maintenance of Myorobots in ways that are not available for other robotic platforms. The flexibility of the design will allow application discovery and improve innovation capacity in a wide range of domains.

We expect to have the most significant impact in the area of robotics R&D, both in academic and industrial settings. In the former we want to enable research in the area of compliant robotics and/or cognitive systems, while in the latter we want to spur innovation in the crucial sector of service robotics. Nevertheless, our target audience also includes educators (using Myorobotics as a teaching platform) and entrepreneurs providing goods and services around the Myorobotics toolkit.

The potential benefits for Europe are significant: improved competitiveness of the European robotics R&D, increased growth in the robotics service sector, new opportunities for service providers and job creation, as well as new and highly competitive consumer and industrial systems.

In our endeavors we plan to involve other researchers and engineers, building a user community that can continue to expand and improve our toolkit long after the project has ended. Indeed, we are already raising awareness by participating in different robotics events in Europe and worldwide, while also planning collaborations with other institutions.

Contact Information:

Dr. Konstantinos Dalamagkidis

Institut für Informatik, I6
Technische Universität München
Parkring 13,
85748 Garching b. München
Germany

Email: dalamagkidis@tum.de

Phone: +49 89 289 17636
Fax: +49 89 289 18107

SEVENTH FRAMEWORK PROGRAMME

The Myorobotics project is funded from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n°288219.

TUM Robotics and Embedded Systems

ETH Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Fraunhofer IPA

brl Bristol Robotics Laboratory