Claims

1. System for controlling actuators of an articulated robot (80) and enabling the robot (80) executing a given task, comprising:

- a first unit (101) providing a specification of robot manipulation skills *s* selectable from a skill space depending on the task, with a robot manipulation skill *s* being defined as a tuple

(*S, O, C_{pre}, C_{err}, C_{suc}, R,* χ_{cmd}*, X, P, Q*) with

S: a Cartesian product of *I* subspaces ζ_i: $S = \zeta_{i=1} \times \zeta_{i=2} \times \ldots \times \zeta_{i=I}$ with *i={1,2,….,I} and* $I \geq 2$*,* wherein the subspaces ζ_i comprise a control variable, in particular a desired variable, or an external influence on the robot or a measured state, in particular an external wrench comprising in particular an external force and an external moment,

O: a set of physical objects,

C_{pre}: a precondition,

C_{err}: an error condition,

C_{suc}: a success condition,

R: nominal result of ideal skill execution,

χ_{cmd}: skill commands,

X: physical coordinates,

P: skill parameters, with *P* consisting of three subsets P_t, P_I, P_D, with P_t being the parameters resulting from a priori knowledge of the task, P_I being the parameters not known initially and need to be learned and/or estimated during execution of the task, and P_D being constraints of parameters P_I,

Q: a performance metric, wherein *Q(t)* is denoting the actual performance of the skill carried out by the robot (80),

- a second unit (102), wherein the second unit (102) is connected to the first unit (101) and further to a learning unit (103) and to an adaptive controller (104),

wherein the adaptive controller (104) receives skill commands χ_{cmd},

wherein the skill commands χ_{cmd} comprise the skill parameters P_I,

wherein based on the skill commands χ_{cmd} the controller (104) controls the actuators of the robot (80),

wherein the actual status of the robot (80) is sensed by respective sensors and/or estimated by respective estimators and fed back to the controller (104) and to the second unit (102),

wherein based on the actual status, the second unit (102) determines the performance value *Q(t)* of the skill carried out by the robot (80), and

wherein the learning unit (103) receives P_D, and *Q(t)* from the second unit (102),

determines updated skill parameters $P_l(t)$ and provides $P_l(t)$ to the second unit (102) to replace hitherto existing skill parameters P_l.

2. System according to claim 1,
wherein a control law is defined as $\tau_u(t) = J(q)^T(-F_{ff} - K(t)e - D\dot{e}) + \tau_r$, where $F_{ff}(t)$ denotes the feed-forward wrench, $K(t)$ a stiffness matrix, D a damping matrix and $J(q)$ a Jacobian, wherein the position and velocity error are denoted by $e = [e_t, e_r]^T$ and $\dot{e} = \dot{x}^* - \dot{x}$, respectively, $e_t = x^* - x$ is the translational position error and $e_r = \theta^* - \theta$ the rotational angle axis error, wherein the dynamics compensator τ_r is defined as $\tau_r = M(q)\ddot{q}^* + C(q,\dot{q})\dot{q}^* + g(q) - sign(\epsilon)\bar{\nu}$, wherein the feed forward wrench F_{ff} is defined as $F_{ff} = F_d(t) + \int_{t_s}^{t} \delta F_{ff} dt + F_{ff,0}$,where $F_d(t)$ is an optional initial time dependent trajectory and $F_{ff,0}$ is the initial value of the integrator, and wherein the adaptive controller (104) adapts feed forward wrench and stiffness via $\delta F_{ff} = F_{ff}(t) - F_{ff}(t-T) = \alpha(\epsilon - \gamma_\alpha(t)F_{ff}(t))$, and $\delta K(t) = K(t) - K(t-T) = \beta(diag(\epsilon \circ e) - \gamma_\beta(t)K(t))$, wherein the adaptive tracking error is defined as $\epsilon = e + \kappa\dot{e}$ with $\kappa > 0$, wherein the positive definite matrices α, β, γ_a and γ_β represent the learning rates for the feed forward and stiffness and the forgetting factors, respectively, and wherein T is the sample time of the controller.

3. System according to one of the claims 1 or 2,
wherein the learning unit (103) carries out a Bayesian and/or a Hierarchical Relative Entropy Policy Search optimization / learning.

4. System according to one of claims 1 to 3,
wherein the system comprises a data interface with a data network, and wherein the system is designed and setup to download system-programs for setting up and controlling the system from the data network.

5. System according to claim 4,
wherein the system is designed and setup to download parameters for the system-programs from the data network.

6. System according to one of the claims 4 to 5,
wherein the system is designed and setup to enter parameters for the system-programs via a local input-interface and/or via a teach-in-process, with the robot (80) being manually guided.

7. System according to one of the claims 4 to 6,

wherein the system is designed and setup such that downloading system-programs and/or respective parameters from the data network is controlled by a remote station, and wherein the remote station being part of the data network.

8. System according to one of the claims 4 to 7,
wherein the system is designed and setup such that system-programs and/or respective parameters locally available at the system are sent to one or more participants of the data network based on a respective request received from the data network.

9. System according to one of the claims 4 to 8,
wherein the system is designed and setup such that system-programs with respective parameters available locally at the system can be started from a remote station, and wherein the remote station is part of the data network.

10. System according to one of the claims 7 to 9,
wherein the system is designed and setup such that the remote station and/or the local input-interface comprises a human-machine-interface HMI designed and setup for entry of system-programs and respective parameters and/or for selecting system-programs and respective parameters from a multitude system-programs and respective parameters.

11. System according to claim 10,
wherein the human-machine-interface HMI is designed and setup such that entries are possible via „drag-and-drop“-entry on a touchscreen, a guided dialogue, a keyboard, a computer-mouse, a haptic interface, a virtual-reality-interface, an augmented reality interface, an acoustic interface, via a body tracking interface, based on electromyographic data, based on elektroenzephalographic data, via a neuronal interface, or a combination thereof.

12. System according to claim 10 or 11,
wherein the human-machine-interface HMI is designed and setup to deliver auditive, visual, haptic, olfactoric, tactile, or electrical feedback or a combination thereof.

13. Robot (80) with a system according to one of claims 1 to 12.

14. Method for controlling actuators of an articulated robot (80) and enabling the robot (80) executing a given task, the robot (80) comprising a first unit (101), a second

unit (102), a learning unit (103), and an adaptive controller (104), the second unit (102) being connected to the first unit (101) and further to a learning unit (103) and to an adaptive controller (104), comprising following steps:

- providing (S1) a specification of robot manipulation skills s selectable from a skill space depending on the task by a first unit (101), with a robot manipulation skill s being defined as a tuple (*S, O, C_{pre}, C_{err}, C_{suc}, R, χ_{cmd}, X, P, Q*) with

S: a Cartesian product of *I* subspaces ζ_i: $S = \zeta_{i=1} \times \zeta_{i=2} \times \ldots \times \zeta_{i=I}$
with *i={1,2,….,I} and* $I \geq 2$, wherein the subspaces ζ_i comprise a control variable, in particular a desired variable, or an external influence on the robot or a measured state, in particular an external wrench comprising in particular an external force and an external moment,
O: a set of physical objects,
C_{pre}: a precondition,
C_{err}: an error condition,
C_{suc}: a success condition,
R: nominal result of ideal skill execution,
χ_{cmd}: skill commands,
X: physical coordinates,
P: skill parameters, with *P* consisting of three subsets P_t, P_I, P_D, with P_t being the parameters resulting from a priori knowledge of the task, P_I being the parameters not known initially and need to be learned and/or estimated during execution of the task, and P_D being constraints of parameters P_I,
Q: a performance metric, while *Q(t)* is denoting the actual performance of the skill carried out by the robot (80),

- an adaptive controller (104) receiving (S2) skill commands χ_{cmd} from a second unit (102),

wherein the second unit (102) is connected to the first unit (101) and further to a learning unit (103) and to the adaptive controller (104) and wherein the skill commands χ_{cmd} comprise the skill parameters P_I,

- controlling (S3) the actuators of the robot (80) by the controller (104) and based on the skill commands χ_{cmd}, wherein the actual status of the robot (80) is sensed by respective sensors and/or estimated by respective estimators and fed back to the controller (104) and to the second unit (102),
- determining (S4) by the second unit (102) and based on the actual status, the

performance value *Q(t)* of the skill carried out by the robot (80),
- the learning unit (103) receiving (S5) P_D and *Q(t)* from the second unit (102), and
- determining (S6) updated skill parameters $P_I(t)$ and providing $P_I(t)$ to the second unit (102) and replacing hitherto existing skill parameters P_I.

15. Computer system with a data processing unit, wherein the data processing unit is designed and set up to carry out a method according to claim 14.

16. Digital data storage with electronically readable control signals, wherein the control signals can coaction with a programmable computer system, so that a method according to claim 14 is carried out.

17. Computer program product comprising a program code stored in a machine-readable medium for executing a method according to claim 14, if the program code is executed on a computer system.

18. Computer program with program codes for executing a method according to claim 14, if the computer program runs on a computer system.