

THE MULTIPLE VIRTUAL END-EFFECTORS APPROACH FOR HUMAN-ROBOT INTERACTION

Agostino De Santis

*PRISMA Lab, Dipartimento di Informatica e Sistemistica
Università degli Studi di Napoli Federico II
Via Claudio 21, 80125 Napoli, Italy
agodesa@unina.it*

Paolo Pierro

*PRISMA Lab, Dipartimento di Informatica e Sistemistica
Università degli Studi di Napoli Federico II
Via Claudio 21, 80125 Napoli, Italy
paolopierro@tin.it*

Bruno Siciliano

*PRISMA Lab, Dipartimento di Informatica e Sistemistica
Università degli Studi di Napoli Federico II
Via Claudio 21, 80125 Napoli, Italy
siciliano@unina.it*

Abstract In this paper, a method for managing redundancy for a mobile robot manipulator is proposed, which is aimed at kinematic control of the system in interaction tasks with humans. The method considers those parts of the manipulator structure —virtual end-effectors (VEEs)— which could potentially hit objects or persons during human-robot interaction. The positioning of each of these various VEEs is considered as a lower-priority task in the inverse kinematics resolution of the robot manipulator, while the order of priorities is dynamically changed during task execution. In addition, it is shown that suitable trajectories are to be planned for VEEs using sensory data, e.g., with potential field methods. A simulation case study for anthropic domains is proposed.

Keywords: Redundancy resolution, physical human-robot interaction, safety, potential fields, obstacle avoidance.

1. Introduction

Human-robot interaction addresses important issues to avoid that the physical body of a robot could result in damages to humans. In the latest years the attention was focused on cognitive aspects of the growing interaction from robots and humans, like mental models and interfaces. It is important to notice that the presence of physical “bodies” is a crucial aspect in the interaction between humans and robots. In particular, physical human-robot interaction (pHRI) addresses the two crucial issues of safety and dependability, especially when environments are unstructured. The physical interaction with a robot in anthropic domains becomes every day more interesting for assistance and service robotics in the houses and for the elderly-dominated society. The EURON project PHRIDOM (Albu-Schaffer et al., 2005), e.g., is addressing these issues. The crucial goals of safety and dependability are related to technical issues such as collision avoidance, redundancy resolution, compliance control and sensory-based safety systems for close interaction.

Safe and dependable interaction can be accomplished both in a passive and in an active fashion. Passive safety is introduced, e.g., using springs, elastic joints (De Luca, 2000); other interesting techniques were also proposed, like the variable-stiffness actuators (Bicchi et al., 2001) and the distributed macro-mini actuation (Zinn et al., 2002). To improve safety, and also to add dependability for the users, active control of the physical interaction is to be considered. Force control (Siciliano and Villani, 1999) and safe postures of robot manipulators should be focused as fundamental issues. In addition, the whole kinematic structure of a manipulator must be controlled, because the robot can hit a person with different parts of the structure.

This paper considers the problem of controlling the positioning of crucial parts of the kinematic structure of a robot in interaction tasks, which are termed “virtual end-effectors” (VEEs). Proper Closed-Loop Inverse Kinematics (CLIK) schemes (Siciliano, 1990) are adopted to achieve resolution in the presence of redundancy, so as to take into account the issues discussed above in the positioning of such VEEs. Each VEE is controlled with a different level of priority with respect to the task, programming the positioning of each dangerous part of the articulated structure in a safe configuration; then, the priorities between the tasks are handled in a hierarchical inverse kinematics scheme (Siciliano and Slotine, 1991). The trajectory planning phase is designed to make the multiple VEEs approach suitable to control of the interaction. In detail, an obstacle avoidance technique based on the well-known potential field

method (Khatib, 1986) is adopted to dynamically change the priority order according to the position of goals and objects in the environment.

2. Modelling

The application domain hereby considered is domestic assistance. For dependable pHRI a redundant mobile robot is needed: movements in a room, objects picking and other tasks may be accomplished, for instance, with a manipulator mounted on a mobile base.

2.1 Kinematics

The mobile robot manipulator considered for the purpose of the present study has the kinematic structure of Fig. 1, which is equivalent to the assembly of a commercial mobile robot (Pioneer PowerBot) and an industrial robot manipulator (Comau Smart-3S), although the method is at all applicable for any kinematic structure with a known Jacobian. In the figure, several critical points are evidenced (A, B, C, D, E), which describe those extremities of the robot that can collide with a human being. Also, they are crucial in order to locate the positions of the manipulator links, since the robot can run into an obstacle not only by a VEE, but also with an intermediate point between two VEEs located on a link.

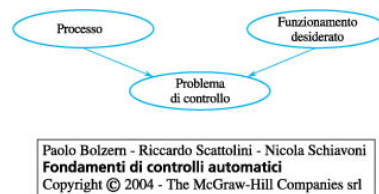


Figure 1. Mobile robot manipulator with VEEs A, B, C, D, E

It should be pointed out, however, that safety issues suggest using accurate sensor information to localize goals and obstacles, lightweight structures and other additional facilities to make the robot intrinsically safe in event of collisions. Here, however, only kinematic aspects are focused. By the way, the manipulator should be lightweight, while industrial manipulators are heavy and cart robots able to carry them are not yet available for potential use in houses.

2.2 Redundancy resolution

Redundancy resolution is related to the problem of finding movements of available joints that respect the desired motion of the end-effector,

while satisfying some additional task. The solution of the problem can be found on the basis of the well-known differential mapping

$$\dot{\mathbf{p}} = \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} \quad (1)$$

where

$$\begin{aligned} \mathbf{p} &= [x \quad y \quad z]^T \\ \mathbf{q} &= [q_1 \quad q_2 \quad \dots \quad q_n]^T \end{aligned}$$

are respectively the end-effector position vector and the joint position vector of an n -DOF mobile robot manipulator, and $\mathbf{J}$ denotes the usual Jacobian. For the purpose of the present work, the end-effector orientation is not considered, while $n = 8$, i.e. 2 DOF's for the mobile base and 6 DOF's for the manipulator. Since the robot is redundant ($n > 3$), the simplest way to invert the mapping (1) is to use the pseudo-inverse of the Jacobian matrix, which corresponds to the minimization of the joint velocities in a least-square sense (Sciavicco and Siciliano, 2000). Because of the different characteristics of the available DOFs, it could be required to modify the velocity distribution. This might be achieved by adopting a weighted pseudo-inverse $\mathbf{J}_W^\dagger$

$$\mathbf{J}_W^\dagger = \mathbf{W}^{-1} \mathbf{J}^T (\mathbf{J} \mathbf{W}^{-1} \mathbf{J}^T)^{-1} \quad (2)$$

with the $(n \times n)$ matrix $\mathbf{W}^{-1} = \text{diag}\{\beta_1, \beta_2, \dots, \beta_n\}$, where β_i is a weight factor belonging to the interval $[0, 1]$ such that $\beta_i = 1$ corresponds to full motion for the i -th degree of mobility and $\beta_i = 0$ corresponds to freeze the corresponding joint (De Santis et al., 2005a).

Redundancy of the system can be further exploited by using a task-priority strategy (Nakamura, 1991) corresponding to a solution to (1) of the form

$$\dot{\mathbf{q}} = \mathbf{J}_W^\dagger(\mathbf{q})\mathbf{v} + \left(\mathbf{I}_n - \mathbf{J}_W^\dagger(\mathbf{q})\mathbf{J}(\mathbf{q}) \right) \dot{\mathbf{q}}_a \quad (3)$$

where $\mathbf{I}_n$ is the $(n \times n)$ identity matrix, $\dot{\mathbf{q}}_a$ is an arbitrary joint velocity vector and the operator $\left(\mathbf{I}_n - \mathbf{J}_W^\dagger \mathbf{J} \right)$ projects the joint velocity vector in the null space of the Jacobian matrix. Also in (3), $\mathbf{v} = \dot{\mathbf{p}}_d + k(\mathbf{p}_d - \mathbf{p})$ which provides a feedback correction term of $\mathbf{p}$ to the desired position $\mathbf{p}_d$, according to the well-known CLIK algorithm, being $k > 0$ a suitable gain (Siciliano, 1990). This solution generates an internal motion of the robotic system (secondary task) which does not affect the motion of the end-effector while fulfilling the primary task.

The kind of secondary tasks employed for the algorithm discussed in this work are based on the inverse kinematics of a reduced part of the

structure. As an example of positioning of different parts of manipulator (rather than only the actual end-effector), consider the human arm: the structure is redundant for the positioning of the hand, and thus it is possible to position the elbow (which can be considered a first VEE); the so-computed joint values can then be used as references for the positioning of the wrist (second VEE), and so far for the hand (real end-effector) (De Santis et al., 2005b). Therefore, a hierarchical solution of redundancy is achieved, where the various lower-priority tasks are to be selected according to some suitable criteria (Featherstone, 1988).

3. The multiple VEEs approach

Virtual end-effectors (VEEs) are parts of the manipulator structure, whose positions are to be controlled in addition to the control of the end-effector of the mobile robot manipulator. In detail, let $\mathbf{q}_i$ denote the vector of the n_i joint variables which determine the position $\mathbf{p}_i$ of the i -th VEE. Therefore, the differential mapping for the VEE is

$$\dot{\mathbf{p}}_i = \mathbf{J}_i(\mathbf{q}_i)\dot{\mathbf{q}}_i \quad (4)$$

where $\mathbf{J}_i$ denotes the associated Jacobian.

The multiple VEEs approach is hereby introduced in a general fashion, by adopting a multiple task priority strategy for specifying secondary tasks, along with a proper trajectory planning technique for the desired motion of each VEE. The result is a nested N -layer CLIK scheme, where N is the number of considered VEEs. To this regard, please notice that the end-effector is included in the counting of the VEEs; in fact, it may well be the case the highest priority be assigned to an intermediate VEE other than to the end-effector, say when an obstacle is obstructing the end-effector motion.

With this approach, the control of different points is not considered in a global matrix, but with multiple mappings. The VEEs approach can be used for maneuvering a kinematic structure in a volume, e.g., for tube inspections and endoscopy with snake robots, by considering the most critical prominences of the structure as VEEs.

3.1 Nested closed-loop inverse kinematics

Inverse kinematics with the VEEs approach orders the VEE positioning tasks according to a priority management strategy. Since the trajectories of lower priority VEEs are assigned as secondary task, they will be followed only if they do not interfere with the higher priority task to be fulfilled. Hence, a list of VEEs is considered, starting from the one with highest priority. When a VEE gets close to an obstacle,

its desired path following (necessary to avoid the obstacle) becomes of higher priority for the CLIK scheme and the priority order is switched with respect to the distance of each VEE from the obstacle. This can be achieved by considering the N -layer priority algorithm described in the following. The idea is summarized in Fig. 2, being N the lowest priority.

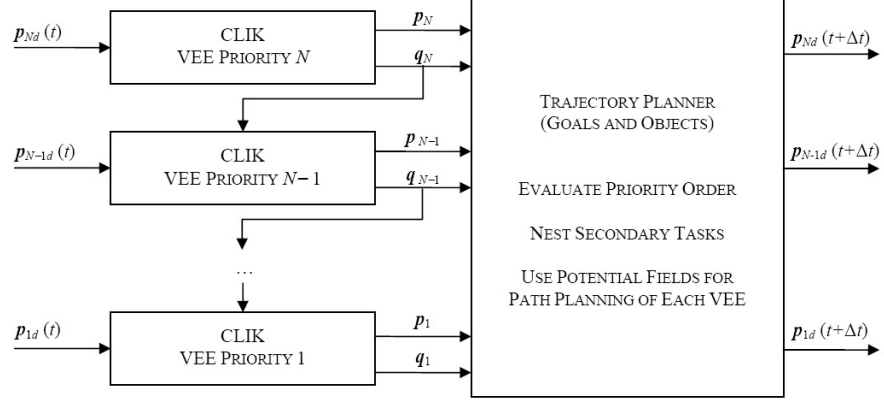


Figure 2. Scheme of nested CLIK with VEEs

At the lowest layer, the differential mapping corresponding to the velocity of the VEE with lowest priority is considered, i.e. (4) with $i = N$. Hence, a CLIK algorithm with weighted pseudo-inverse is adopted to compute the inverse kinematics:

$$\dot{\mathbf{q}}_N = \mathbf{J}_N^\dagger(\mathbf{q}_N)\mathbf{v}_N, \quad (5)$$

with $\mathbf{v}_N = \dot{\mathbf{p}}_{Nd} + k_N(\mathbf{p}_{Nd} - \mathbf{p}_N)$, being $\mathbf{p}_{Nd}$ the desired position for the VEE with lowest priority and $k_N > 0$. The pseudo-inverse matrix is substantially weighted as in (2).

At the generic i -th layer of the nested scheme, with $i = N - 1, \dots, 1$, the inverse kinematics is computed as in (3), i.e.

$$\dot{\mathbf{q}}_i = \mathbf{J}_{iW}^\dagger(\mathbf{q}_i)\mathbf{v}_i + \left(\mathbf{I}_{n_i} - \mathbf{J}_{iW}^\dagger(\mathbf{q}_i)\mathbf{J}_i(\mathbf{q})\right)\dot{\mathbf{q}}_{ia} \quad (6)$$

where

$$\mathbf{J}_{iW}^\dagger = \mathbf{W}_i^{-1}\mathbf{J}_i^\mathrm{T}(\mathbf{J}_i\mathbf{W}_i^{-1}\mathbf{J}_i^\mathrm{T})^{-1} \quad (7)$$

with $\mathbf{W}_i^{-1} = \text{diag}\{\beta_{i1}, \beta_{i2}, \dots, \beta_{in_i}\}$ and $\mathbf{v}_i = \dot{\mathbf{p}}_{id} + k_i(\mathbf{p}_{id} - \mathbf{p}_i)$, being $\mathbf{p}_{id}$ the desired position for the VEE with priority i , and $k_i > 0$. Further, $\dot{\mathbf{q}}_{ia}$ is the gradient of the objective function:

$$G_i = -\frac{1}{n_i} \sum_{j=1}^{n_i} \frac{q_{i,j} - q_{i+1,j}}{q_{i,jM} - q_{i,jm}}$$

where $q_{i,jM}$ ($q_{i,jm}$) is the maximum (minimum) value of the joint variable $q_{i,j}$, i.e. the j -th component of the joint vector $\mathbf{q}_i$. The above choice corresponds to achieving a joint motion for the joint variables as close as possible to that computed in the previous layer $\mathbf{q}_{i+1}$, which are feeded as secondary reference values to the next layer, providing a way to fulfill inverse kinematics of different parts of the structure, with different priorities.

Going on, the nested CLIK algorithm computes the inverse kinematics for the structures ending with each of the considered VEEs. It is worth emphasizing that the order of priorities is dynamically changed during task execution. As illustrated in Fig. 2, the scheme takes the desired positions for the ordered VEEs at time t ; then, the output of the CLIK algorithms at the N levels are input to a trajectory planning block which re-evaluates the priority order among the various VEEs according to the positions of the goals and the obstacles in the environment, and thus generates the new ordered sequence of desired positions for the VEEs at time $t + \Delta t$, where Δt is the sampling time at which the CLIK algorithm is discretized for practical implementation; the planning aspects are discussed in the following section.

The weights β_{ij} of the matrices $\mathbf{W}_i$ are chosen according to the criterion described above (De Santis et al., 2005a). In particular, the joints of the mobile base have a higher weight with the respect to the joints of the manipulator. In Fig. 3 it is possible to see how trajectories are followed for various VEEs with different priorities. It is easy to recognize that planned trajectories for VEEs with lower priorities are abandoned when they interfere with the higher priority planned trajectories. With the priority strategy, which ranks the VEEs, it is possible to fulfill the most critical positioning problems on line.

The time history of a joint variable is also reported in Fig. 4; movements planned in the task space are abandoned if they interfere with higher-priority tasks.

The priority management strategy is crucial: if a mobile robot manipulator is considered, which avoids the head of a person in a room with the end-effector, it is difficult to predict the possibility of hitting him/her with other parts of the structure, because a possible avoidance movement can be incompatible with the path of the real end-effector or of a VEE with higher priority.

3.2 Trajectory planning

For each VEE it should be possible to plan a complete trajectory offline, but this approach is not satisfying, since the trajectories imposed

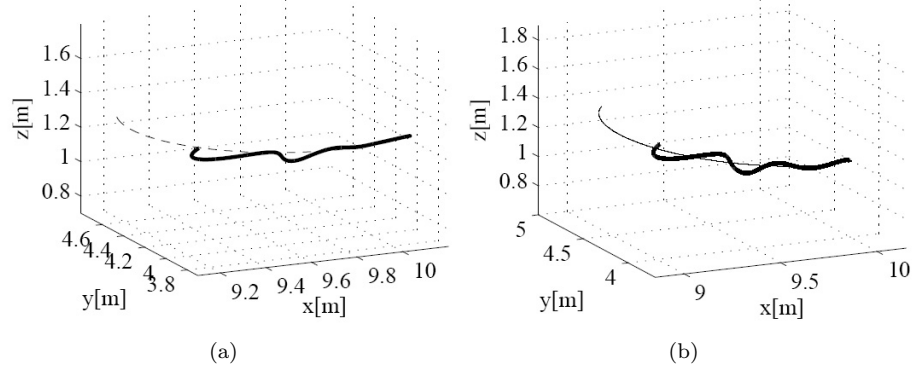


Figure 3. Planned (gray) and actual (black) trajectory for the VEE labelled with A and B in Fig. 1

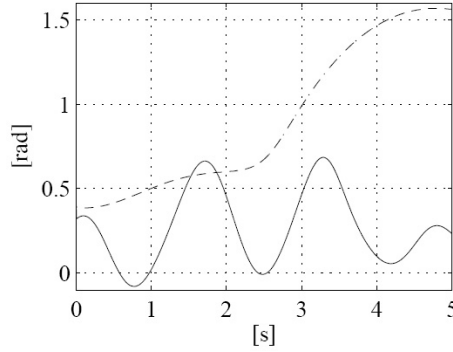


Figure 4. Planned (dashed) and actual (solid) trajectory for the joint value q_4

to the VEEs, if planned a priori, do not have realistic application (most obstacles are moving and their trajectories are not known a priori). Furthermore, the positioning of VEEs in task space is only a secondary task for the positioning of other VEEs with higher priority. On-line trajectory planning is needed: potential fields methods (Khatib, 1986) can be used for such a planning: the manipulator moves in a field of forces; the position to be reached is an attractive pole for the end-effector, and obstacles are repulsive surfaces for the manipulator parts. The field of artificial forces F is produced by a differentiable potential function, with $F(\mathbf{p}_i) = -\nabla(U(\mathbf{p}_i))$, where $\nabla(U(\mathbf{p}_i))$ is the gradient vector of U at $\mathbf{p}_i$, that is the position of the VEE under the field effect.

The potential U is constructed as the sum of two elementary potential functions: $U(\mathbf{p}_i) = U_{att}(\mathbf{p}_i) + U_{rep}(\mathbf{p}_i)$. In this relation U_{att} is the attractive potential and depends only on the final position, whereas U_{rep} is the

repulsive function and depends only on the obstacles position. During the simulation, the attractive potential field is chosen to be a parabolic well, centered in the goal positions, whereas a repulsive field is related to a distance of influence from obstacles. So, the goal is a source of an attractive potential field; obstacles are sources of repulsive potential fields.

Simulations of robot movements with this approach are shown in the next section. Notice again the importance of the priority management in the CLIK schemes, as discussed above: even with on-line path planning, a desired path could be not executable, so it is necessary to switch the priority order, as emphasized by the trajectory planning block in Fig. 2

4. A case study

In this section, operations of the considered mobile manipulator are examined in an anthropic domain, while the robot is performing an assistance task. The model has been simulated under the Virtual Reality Toolbox (VRT) in Matlab. The VRT is a solution for interacting with virtual reality models of dynamic systems over time. The complete CLIK algorithm with multiple priority handling and potential fields method for trajectory planning is considered. The robot has to take a soda can which is placed in a fitted cupboard, where an impaired person cannot arrive. In the first phase, the person on the wheelchair is a moving obstacle to be avoided. When the robot takes the can, the person becomes a goal point to be reached by the end-effector bringing the can. The first desired position for the robot is the location of the can on the cupboard; after the end of the goal reaching, the second desired position is the location of the person, who is moving in an independent way. The anthropic environment where the task is executed is shown in Fig. 5.

In the shown environment, the robot has to avoid the lamps (it has to slow down in order to keep away from objects attached to the ceiling), as well as the person (turning around him/her) and finally the desktop and the table. The obstacles are modelled as spheres: a lamp is a sphere having a small radius, whereas a person is a set of spheres with different radiuses. Notice that these radiuses may take into account cognitive evaluations. As an example, consider the potentially unpredictable behaviours of children in the scene: in such a case, safety issues suggest to magnify the radiuses for repulsive potential fields. It is possible, in intelligent environments, to provide radio-frequency identification tags to assign different set of parameters, depending on the skills of interacting people, for robot kinematic and force control. The table has been mod-



Figure 5. Interaction with the service robot in an anthropic environment

elled differently: it is characterized by a succession of points (each point represents a corner or the lower point of a foot of the table). The robot is equipped with seven leds, each one placed next to one critical points: when a particular VEE assumes the highest priority, its led lights up. The priority assignment for the nested CLIK of the different VEEs is reported in Fig. 6, where EE stands for the end-effector and WMR for the wheeled-mobile robot.

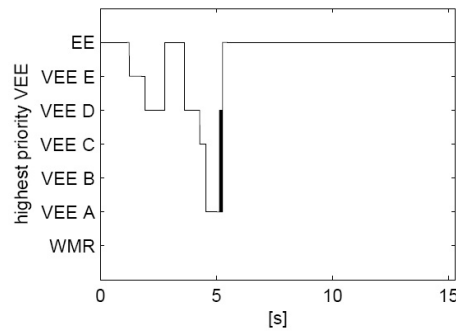


Figure 6. Time history of priority assignments

In Fig. 7 it is possible to see the paths for one VEE and the EE avoiding the obstacles as well as the motion of the impaired person. Strong hypothesis about sensory system are assumed: positions of robot, people and objects are exactly known and tracked.

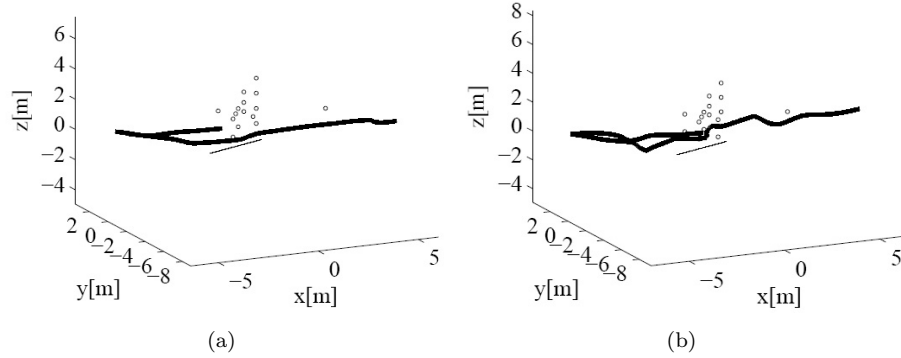


Figure 7. Trajectories for the VEE labelled with A and the EE of the robot (solid) in the task of Fig. 5, where the spheres are edges of obstacles and the thin line is the trajectory of the impaired person during the task

5. Conclusion

The virtual end-effectors (VEEs) approach has been proposed for simultaneous inverse kinematics, with proper trajectory planning, of mobile robot manipulators operating in unstructured environments. For pHRI, the VEEs approach can actually complement interaction schemes, where intrinsical and active safety issues are to be considered. The proposed approach may be also used to keep a robot in proper postures in narrow spaces (e.g., snake robots) and for self-collision avoidance too. For comparison, a global optimization of different forces acting on the manipulator without null-space techniques should also be considered, with a weighted extended Jacobian approach. In addition, automatic techniques for the location of the VEEs should be of interest as well. Future work will also be devoted to add soft-computing techniques for both trajectory planning and inverse kinematics, and to consider integration with force control on real mobile robot manipulators.

References

- Albu-Schaffer, A., Bicchi, A., Boccadamo, G., Chatila, R., De Luca, A., De Santis, A., Giralt, G., Hirzinger, G., Lippiello, V., Mattone, R., Schiavi, R., Siciliano, B., Tonietti, G., Villani, L., “Physical Human-Robot Interaction in Anthropic Domains: Safety and Dependability”, *4th IARP/IEEE-EURON Workshop on Technical Challenges for Dependable Robots in Human Environments*, Nagoya, J, July 2005.
- De Luca, A., “Feedforward/feedback laws for the control of flexible robots” *2000 IEEE International Conference of Robotics and Automation*, San Francisco, CA, USA, April 2000.

- Bicchi, A., Tonietti, G., Bavaro, M., Piccigallo, M., "Variable stiffness actuators for fast and safe motion control", *11th International Symposium of Robotics Research*, Siena, I, October 2003.
- Zinn, M., Khatib, O., Roth, B., Salisbury, J.K., "A new actuation approach for human friendly robot design", *International Symposium on Experimental Robotics*, S. Angelo d'Ischia, I, July 2002.
- Siciliano, B., Villani, L., *Robot Force Control*, Kluwer Academic Publishers, Boston, MA, 1999.
- Siciliano, B., "A closed-loop inverse kinematic scheme for on-line joint-based robot control", *Robotica*, 8, 231–243, 1990.
- Sciavicco, L., Siciliano, B., *Modelling and Control of Robot Manipulators*, (2nd Ed.), Springer-Verlag, London, UK, 2000.
- Siciliano, B., Slotine, J.J.E., "A general framework for managing multiple tasks in highly redundant robotic systems", *5th International Conference on Advanced Robotics*, Pisa, I, June 1991.
- Khatib, O., "Real-time obstacle avoidance for robot manipulators and mobile robots", *International Journal of Robotics Research*, 5(1), 90–98, 1986.
- De Santis, A., Siciliano, B., Villani, L., "Fuzzy trajectory planning and redundancy resolution for a fire fighting robot operating in tunnels", *2005 IEEE International Conference on Robotics and Automation*, Barcelona, E, April 2005.
- Nakamura, Y., *Advanced Robotics: Redundancy and Optimization*, Addison-Wesley, Reading, Mass., 1991.
- De Santis, A., Caggiano, V., Siciliano, B., Villani, L., Boccignone, G., "Anthropic inverse kinematics of robot manipulators in handwriting tasks", *12th Conference of the International Graphonomics Society*, Fisciano, Italy, June 2005.
- Featherstone, R., "Resolving manipulator redundancy by combining task constraints", *Int. Meeting Advances in Robot Kinematics*, Ljubljana, Yugoslavia, Sep. 1988.