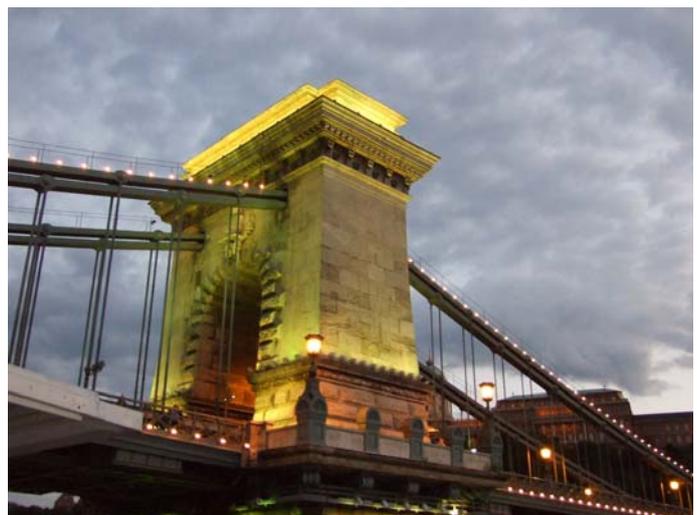


ISSN 1590-8844  
*Vol. 12 No 01*  
2011

# *International Journal of Mechanics and Control*

**Editor: Andrea Manuello Bertetto**



LIBRERIA EDITRICE UNIVERSITARIA  
**LEVROTTO & BELLA**  
TORINO

Editorial Board of the  
***International Journal of Mechanics and Control***

Published by Levrotto&Bella – Torino – Italy E.C.

*Honorary editors*

**Guido Belforte**

**Kazy Yamafuji**

**Editor: Andrea Manuello Bertetto**

**General Secretariat: Elvio Bonisoli**

Atlas Akhmetzyanov  
*V.A.Trapeznikov Institute of Control Sciences  
of Russian Academy of Sciences  
Moscow – Russia*

Domenico Appendino  
*Prima Industrie  
Torino – Italy*

Kenji Araki  
*Saitama University  
Shimo Okubo, Urawa  
Saitama – Japan*

Guido Belforte  
*Technical University – Politecnico di Torino  
Torino – Italy*

Bruno A. Boley  
*Columbia University,  
New York – USA*

Marco Ceccarelli  
*LARM at DIMSAT  
University of Cassino  
Cassino – Italy*

Amalia Ercoli Finzi  
*Technical University – Politecnico di Milano  
Milano – Italy*

Carlo Ferraresi  
*Technical University – Politecnico di Torino  
Torino – Italy*

Anindya Ghoshal  
*Arizona State University  
Tempe – Arizona – USA*

Nunziatino Gualtieri  
*Space System Group  
Alenia Spazio  
Torino – Italy*

Alexandre Ivanov  
*Technical University – Politecnico di Torino  
Torino – Italy*

Giovanni Jacazio  
*Technical University – Politecnico di Torino  
Torino – Italy*

Takashi Kawamura  
*Shinshu University  
Nagano – Japan*

Kin Huat Low  
*School of Mechanical and Aerospace Engineering  
Nanyang Technological University  
Singapore*

Andrea Manuello Bertetto  
*University of Cagliari  
Cagliari – Italy*

Stamos Papastergiou  
*Jet Joint Undertaking  
Abingdon – United Kingdom*

Mihailo Ristic  
*Imperial College  
London – United Kingdom*

János Somló  
*Technical University of Budapest  
Budapest – Hungary*

Jozef Suchy  
*Faculty of Natural Science  
Banska Bystrica – Slovakia*

Federico Thomas  
*Instituto de Robótica e Informática Industrial  
(CSIC-UPC)  
Barcelona – Espana*

Lubomir Uher  
*Institute of Control Theory and Robotics  
Bratislava – Slovakia*

Furio Vatta  
*Technical University – Politecnico di Torino  
Torino – Italy*

Vladimir Viktorov  
*Technical University – Politecnico di Torino  
Torino – Italy*

Kazy Yamafuji  
*University of Electro-Communications  
Tokyo – Japan*

*Official Torino Italy Court Registration  
n.5390, 5<sup>th</sup> May 2000*

*Deposito presso il Tribunale di Torino  
numero 5390 del 5 maggio 2000*

*Direttore responsabile:*

*Andrea Manuello Bertetto*

# ***International Journal of Mechanics and Control***

***Editor: Andrea Manuello Bertetto***

***Honorary editors: Guido Belforte  
Kazy Yamafuji***

***General Secretariat: Elvio Bonisoli***

The Journal is addressed to scientists and engineers who work in the fields of mechanics (mechanics, machines, systems, control, structures). It is edited in Turin (Northern Italy) by Levrotto&Bella Co., with an international board of editors. It will have not advertising.

Turin has a great and long tradition in mechanics and automation of mechanical systems. The journal would will to satisfy the needs of young research workers of having their work published on a qualified paper in a short time, and of the public need to read the results of researches as fast as possible.

Interested parties will be University Departments, Private or Public Research Centres, Innovative Industries.

## ***Aims and scope***

The *International Journal of Mechanics and Control* publishes as rapidly as possible manuscripts of high standards. It aims at providing a fast means of exchange of ideas among workers in Mechanics, at offering an effective method of bringing new results quickly to the public and at establishing an informal vehicle for the discussion of ideas that may still in the formative stages.

## ***Language: English***

*International Journal of Mechanics and Control* will publish both scientific and applied contributions. The scope of the journal includes theoretical and computational methods, their applications and experimental procedures used to validate the theoretical foundations. The research reported in the journal will address the issues of new formulations, solution, algorithms, computational efficiency, analytical and computational kinematics synthesis, system dynamics, structures, flexibility effects, control, optimisation, real-time simulation, reliability and durability. Fields such as vehicle dynamics, aerospace technology, robotics and mechatronics, machine dynamics, crashworthiness, biomechanics, computer graphics, or system identification are also covered by the journal.

*Please address contributions to*

Prof. Guido Belforte  
Prof. Andrea Manuello Bertetto  
PhD Eng. Elvio Bonisoli

*Dept. of Mechanics  
Technical University - Politecnico di Torino  
C.so Duca degli Abruzzi, 24.  
10129 - Torino - Italy - E.C.*

www.jomac.it  
e\_mail: jomac@polito.it

## *Subscription information*

Subscription order must be sent to the publisher:

*Libreria Editrice Universitaria  
Levrotto&Bella  
2/E via Pigafetta – 10129 Torino – Italy*

www.levrotto-bella.net  
e\_mail: info@levrotto-bella.net  
tel. +39.011.5097367  
+39.011.5083690  
fax +39.011.504025



# THE DEVELOPMENT OF A MEMS/NEMS-BASED 3 D.O.F. COMPLIANT MICRO ROBOT

Marco Balucani\*    Nicola Pio Belfiore\*\*    Rocco Crescenzi\*    Matteo Verotti\*\*

\* Sapienza University of Rome, Department of Electronic Engineering

\*\* Sapienza University of Rome, Department of Mechanical and Aerospace Engineering

## ABSTRACT

Microrobots are used nowadays in several fields of application, especially in mini invasive surgery. However, they are rather difficult to be constructed, and the traditional machining tools are not adequate to build their smaller parts. The construction of microrobots is even harder if more than one D.O.F. are required, because these systems are more complicated. This paper deals with the development of a 3 D.O.F. planar micro platform with a remote system of actuation. The new approach of design and manufacturing is based on two innovative solutions: a) the adoption of the technologies used to build MEMS, Micro Electro Mechanical Systems; b) the introduction of new flexural hinge to develop compliant micro mechanisms. The new concept of flexural hinge is described in the paper, also from a theoretical point of view. Several examples of possible structures are, then, proposed and analyzed, together with their remote wire actuation systems. Finite Element Analysis (FEA) has been also adopted to analyze the system performance under small deformations and large deflections. The principle of fabrication is, finally, described. The process consists of a sequence of single steps which have allowed to achieve an overall maximum size down to 3–4 mm and the minimum thickness of the smaller components down to 50  $\mu\text{m}$ .

Keywords: MEMS, micro robot, compliant mechanisms.

## 1 INTRODUCTION

Robot miniaturization is a recent topic that has been studied with interest for various applications. Among them, mini invasive surgery or diagnostic seem very prolific fields for micro and nano robots. However, the practical construction of such devices is quite complicated.

Robots equipped with the traditional kinematic pairs present the problem of manufacturing and assembling small parts. Such parts can be much smaller than the maximum overall dimension, let us say two orders of magnitude, and so, for example, a system which appears as a mini robot is actually a micro robot, or an apparent micro robot can be actually a nano robot.

The classical methods based on standard machining present some serious limits for the development of micro robots. Surfaces are poorly finished, the material is subject to involuntary treatments, and some resistance problems may also arise, depending on the material used.

For these reasons the Authors decided to start a research program which includes the construction of small robots with maximum overall size in the mm range, by using the MEMS technology.

### 1.1 A BRIEF REVIEW ABOUT THE ADOPTED TECHNOLOGY

The MEMS technologies are usually divided in surface and bulk micromachining. The surface micromachining, mainly processing deposition and etching of different structural materials on top of the substrate (e.g. silicon), reaches a total thicknesses for the material not higher than tens of microns and such thickness is not considered enough in order to realize a compliant Micro Robot that can have a significant application area. In fact, the interaction forces

---

Contact authors: Marco Balucani<sup>1</sup>, Nicola Pio Belfiore<sup>2</sup>

Via Eudossiana, 18, 00184 Rome, Italy

<sup>1</sup>Email: balucani@die.uniroma1.it

<sup>2</sup>Email: belfiore@dima.uniroma1.it

are expected to reach even the order of units of N. The suitable technology for a micro robot must give the possibility to work with at least hundreds of microns in thickness for the material and, at the same time, to be able to use the resolution of the microelectronic lithography technique (e.g. 1 micron or better).

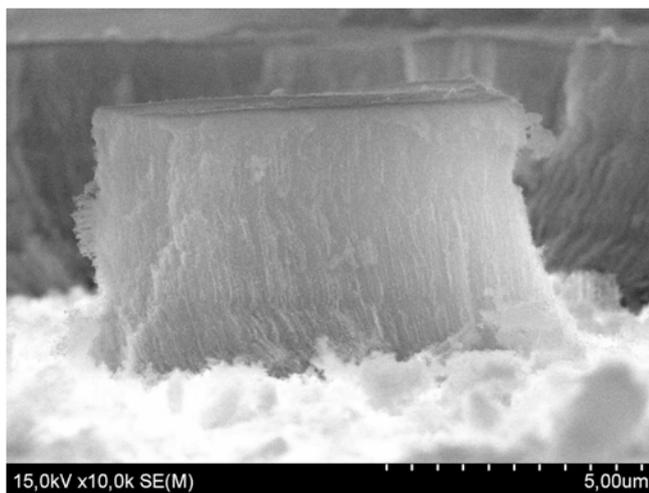


Figure 1 A SEM view of the Ni nanowires after the removal of the porous silicon host material.

The MEMS silicon bulk micromachining offers such opportunity. Moreover, the possibility of integrating MEMS bulk micromachining with Nanotechnology allows to realize special nano-composite materials with unique mechanical, electrical and/or thermal properties. The possibility to deeply etch silicon by plasma and/or wet silicon etching technique allows to realize a thick mold stamp made of silicon where is possible to cast thermosetting polymer. Furthermore, it is possible to fill the polymer by nanoparticles/nanowires in order to modify the mechanical properties or with the inclusion of magnetic nanoparticles/nanowires it is possible to make the polymer sensitive to magnetic field for example generated by a micro-coil. This work represents a first report about the project of construction of a 3 D.O.F. Compliant Micro Robot by using nanostructured silicon (i.e. porous silicon). According to this method, it is possible to realize a 3D mold defined by the standard microelectronic lithography technique in which a thermosetting polymer is casted. By filling partially the pores with a magnetic material (e.g. nanowires and/or nanoparticles) and then by casting the polymer, it is possible to obtain nanocomposite material with magnetic properties that integrated with a micro-coil could be used as actuator for the 3 D.O.F. micro robot. At present, the research group is building the first prototype samples of micro manipulators. All the technological procedures can be actuated with no problems. A first example of Ni nanowires has been obtained by let it grow in silicon porous structures. The corresponding cleaned

actual nanowires have been obtained by removing the silicon structures, as it appears as in Figure 1.

As far as micro-actuation is concerned, the research group is adopting a proprietary technology described in [5] and [6]. Figure 2 shows an example of micro-coil entirely developed by the team.

## 2 A NEW FLEXURAL HINGE

Compliant mechanisms have been used in the recent past for many purposes and they have been appreciated for several reasons. Among them the following are the most interesting:

- there are no sub-parts of the system, which consists of a whole block;
- they can be built by using a single piece of material;
- they have a stable undeformed position which can be used as the neutral reference position;
- mechanical backlash is absent;
- they can be studied in several ways, using for instance non linear FEA, or the classic mechanical theory based on continuum mechanics;
- pseudo-rigid-body equivalent mechanisms can be used to analyze, in a first approximation, the relative displacements between the most significant parts of the compliant system;
- they can be actuated in many different ways, including wire remote actuation.



Figure 2 An example of thick metal micro-coils obtained by using [5] and [6].

These advantages could be very useful also in the field of micro-systems. In fact, they seem very well adaptable to an approach of construction based on the monolithic one-step process of construction. However, the miniaturization of this class of mechanisms is afflicted by some serious problems, such as, for example, the extreme reduction of the smaller sections.

As known, compliant mechanisms work under the principle that there are two types of sub-parts in the systems:

- deformable sub-parts with reduced sections;
- pseudo-rigid sub-parts.

The latter correspond to the links of a traditional mechanism, whose motion is provided by the kinematic pairs. The deformable sub-parts are those on which the mobility of the system relies. Such portions can be much smaller than the others that represent the pseudo-rigid bodies. In [1], for example, an in-plane rotary bi-stable micro-mechanism is illustrated where a tether has  $5.3\ \mu\text{m}$  of width, while the body size of the other parts are about  $500\ \mu\text{m}$ .

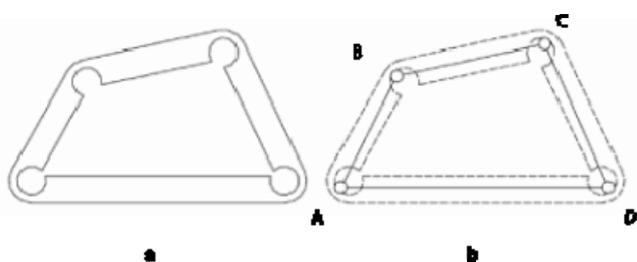


Figure 3 The classical example of a compliant mechanism a) and its corresponding pseudo-rigid-body mechanism b).

The pseudo-rigid body equivalent mechanism can be obtained by substituting the pseudo-rigid sub parts with rigid bodies and by introducing joints in a position which represents, as better as possible, the center of the relative motion between the connected bodies. The classical example is depicted in Figure 3, where the four bar linkage  $ABCD$  is the pseudo-rigid body equivalent mechanism. The revolute pair  $A$  is positioned, for example, as much near as possible to the point which corresponds to the center of relative rotation between the links  $AB$  and  $DA$ .

Unfortunately, such center of the relative rotations is not constant in the plane, because it depends on the intensity of the relative deflections, which are rather larger than in the other structures used in traditional engineering. Hence, the non-linear theories should be adopted to evaluate the relative displacements and deformations. This makes any pseudo-rigid body equivalent mechanisms a prediction model which is not much accurate because it suffers from errors that cannot be neglected in the general case.

Another approach is represented by the topology optimization technology, which can be used to obtain an optimal mechanism layout achieving the best overall system performance. One recent explanation of this method is reported in [4], where one example of compliant MEMS mechanism has been proposed with the aid of integrated topology and shape optimization.

Finally, distributed compliance which employs flexural links, as described in [3], is a promising approach for designing compliant mechanisms for MEMS applications.

In the present paper, an alternative solution to the traditional compliant mechanism is proposed, as previously

registered in the patent [7], where a new flexural hinge is illustrated. A basic principle is therein adopted to introduce equivalent revolute pairs into a compliant mechanisms, namely, that compliance provides (limited) mobility, while geometry provides accuracy.

The new idea is based both on the selective compliance of sub-parts with smaller sections (flexible joints), and on the actions of classic conjugate surfaces (kinematic elements). With reference to Figure 4, joint  $k$  represents a flexible subpart delimited by the extreme sections  $j-k$  and  $i-k$ . Blocks  $i$  and  $j$  represent the two pseudo-rigid links that compose the revolute pair  $i-j$ . The interface  $i-j$  consists of a portion of a cylindrical surface whose trace in the plane is the circular arc represented in the Figure as the conjugate surface  $i-j$ . Finally, the center of the conjugate arc is positioned within the small region where the center of relative rotation of the sections  $j-k$  and  $i-k$  is likely to be, for the allowed deflection status. It can be shown that this point can be placed, with a good approximation, in the center of the elastic weights of the deformable beam. In the Figure, point  $C$  represents a possible choice for that example.

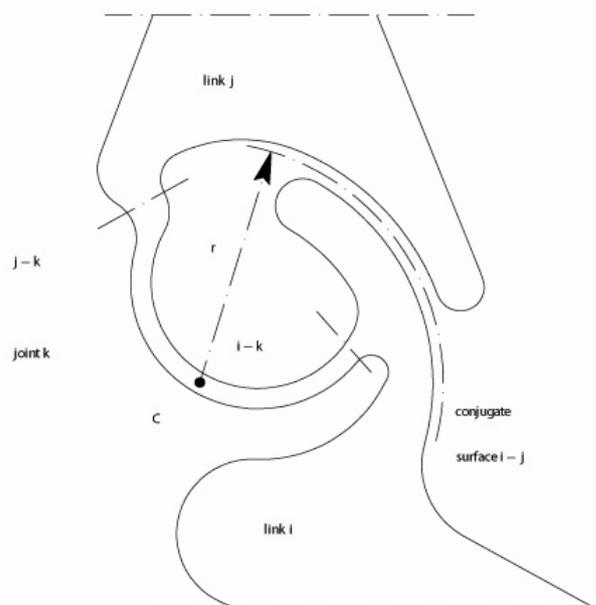


Figure 4 Geometry of the revolute pair.

This approach allows to improve accuracy in design because the centers of the revolute pair conjugate surfaces, as for the pair  $i-j$  in the Figure, are coincident with the relative rotation centers between the pseudo-rigid bodies, as the links  $i$  and  $j$  in the Figure. For the case study reported in the Figure, the elastic beam, represented by joint  $k$ , does not add theoretically any other constraint to the structure. Of course, this result cannot be maintained for the whole set of deformations because the position of the center of relative rotations between the extreme sections of the beam is not constant. However, the choice of the center of the beam

elastic weights guarantees, at least for small deflections, that the over constraints due to the contemporary presence of the interface and the elastic beam are minimized.

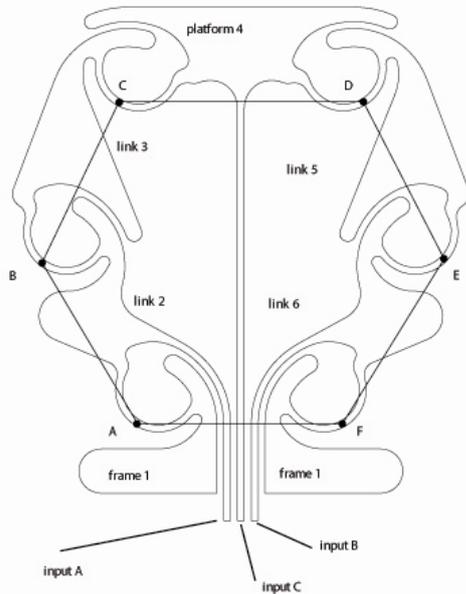


Figure 5 Geometry of the first model of 3 D.O.F. planar platform.

### 3 A NEW METHOD FOR DEVELOPING COMPLIANT PLANAR 3 D.O.F. PLATFORMS

Based on the ideas reported in the previous paragraph, new class of mechanisms can be designed by modifying any classical articulated mechanisms, simply by introducing a flexural hinge, as reported in [7], in place of each revolute kinematic pair. In this paper, we will focus our attention only of compliant planar 3 D.O.F. platforms which will make use of the new flexural hinges.

This idea, which has been better described in [8], can be illustrated by observing the 6 bar linkage  $ABCDEF$  depicted in Figure 5, which can be assumed as the pseudo-rigid-body mechanisms of the compliant platform composed of the frame link 1, the binary chains 2–3 and 6–5 and the platform 4. The latter has 3 degrees of mobility with respect to the frame link, since the elastic joints are replaced by revolute pairs. In fact, Grübler's formula in the plane yields

$$F = \lambda(n - 1) - 2 \cdot 6 = 15 - 12,$$

where  $\lambda$  is the mobility number and  $n$  is the number of rigid links.

Four different sets of layouts have been conceived, each one with a different remote wire actuation method.

### 3.1 A BRIEF NOTE ON THE EVALUATION OF THE D.O.F.

Since the developed microrobots are deformable, there are infinite degrees of freedoms. However, it can be quite satisfying to evaluate the D.O.F. of the corresponding pseudo-rigid-body equivalent mechanism.

In compliant mechanisms, the number of independent actuators can be different from the number of D.O.F. In fact, the existence of the neutral configuration would allow also a null number of D.O.F. for simply static use, losing any control capacity. By using only one actuator on the structure, the manipulator would assume only one sequence of configurations, and the only control variable would be the force intensity at the wire. In any case, if the number of actuators is less than the number of D.O.F., the platform cannot be perfectly under control, because there would be less free parameters than required in order to define the configuration. Finally, a number of wire actuators greater than the D.O.F. would introduce a certain grade of redundancy in actuation and control, but this wouldn't be a problem since the simultaneous action of the actuators could be optimized and redundancy could be used to increase precision.

## 4 PRESENTATION OF THE FOUR GROUPS WITH DIFFERENT GEOMETRY

The first two groups of structures have been conceived with the aim of including, within the constructing process of the robots, all the wires, as sub-parts of the unique block which forms the compliant micro-robot. Therefore, for these cases, there is no need to attach any wire to the corresponding pseudo-rigid links, simply because the wire is already part of the pseudo-rigid link. In the third and fourth groups, the structures are built without including the wires, so postponing the procedure of wire attaching later. This procedure adds another phase to the whole process but has the advantage of simplifying the former step.

### 4.1 THE FIRST GROUP

The first group of platforms is represented in Figure 6, from a) to d) with different sizes. For this group there are 3 actuating wires, two attached to the base hinges and one directly attached to the platform.

The central wire can control the downward translation of the platform, while the upward direction is obtained by pulling both the lateral wires. Platform can rotate by pulling only one of the wires attached on the base hinge

### 4.2 THE SECOND GROUP

The second group of platforms is represented in Figure 7, from a) to d) with different sizes. For this group there are 4 actuating wires, one more with respect to the previous

group. There is one grade of redundancy, which can be managed by the control algorithm by minimizing, for the desired configuration, the sum of the wire tensions. In this group of structures, there are two wires attached to the base hinges and two that are attached on the links that are adjacent to the platform. Platform translations are obtained by pulling selectively one pair of symmetric wires, for example, as in the previous case, the upward translation is obtained by pulling both the lateral wires which actuate the base hinges. Platform rotations are obtained by pulling the wires asymmetrically.

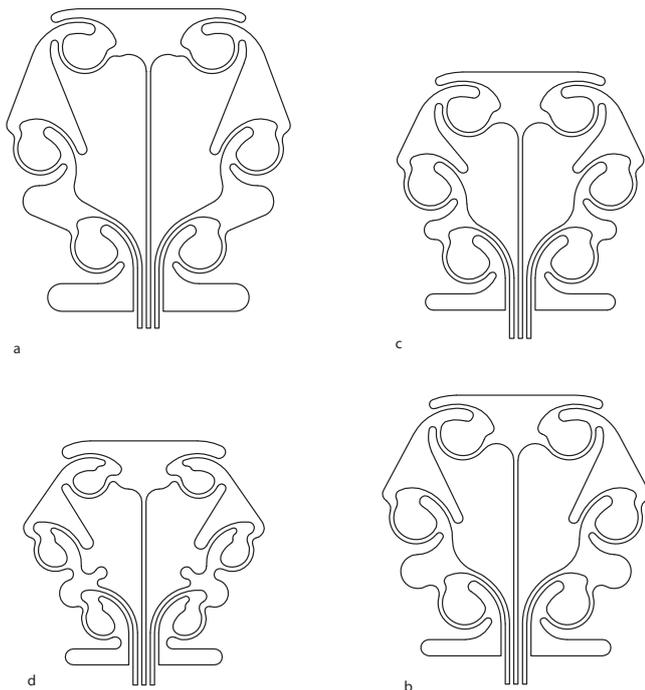


Figure 6 Geometries of 3 D.O.F. planar platforms belonging to the first group: from a) to d), models with 3 cables and decreasing overall dimensions.

### 4.3 THE THIRD GROUP

The third group of platforms is represented in Figure 8, from a) to d) with different sizes. For this group there may be from 0 to 5 actuating wires, depending on the wire attached in corresponding of the circular attaching zones. One cable is attached directly on the platforms, which will make downward translations easier. Redundancy will depend on the choice of the number of attached wires. Rotations and translations can be obtained by actuating, as for the previous groups, the wires either symmetrically or asymmetrically.

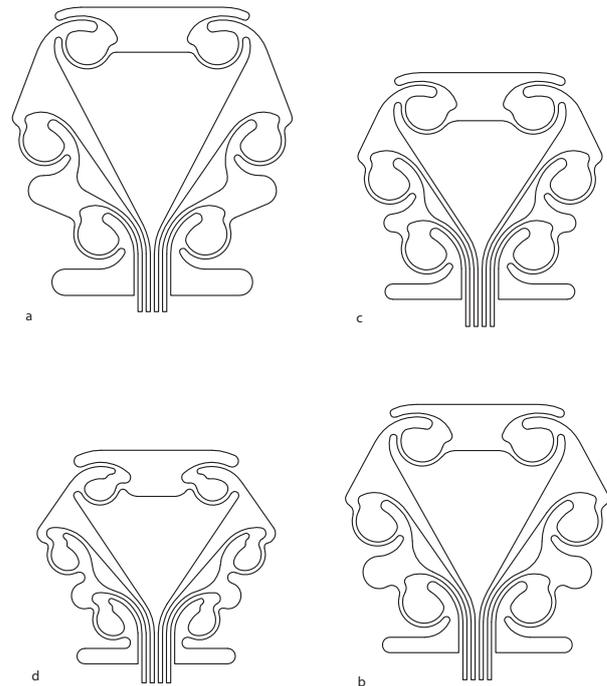


Figure 7 Geometries of 3 D.O.F. planar platforms belonging to the second group: from a) to d), models with 4 cables and decreasing overall dimensions.

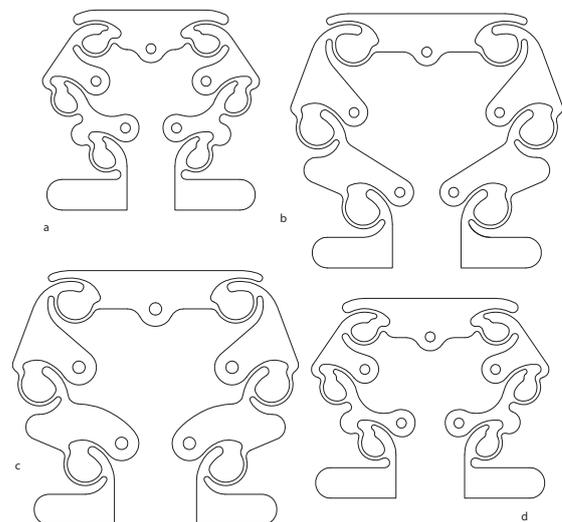


Figure 8 Geometries of 3 D.O.F. planar platforms belonging to the third group: from a) to d), models with decreasing overall dimensions and a different wire attaching method.

#### 4.4 THE FOURTH GROUP

The fourth group of platforms is represented in Figure 9, from a) to d) with different sizes. For this group there may be from 0 to 6 actuating wires, depending on the wire attached in corresponding of the circular attaching zones. Two cables are attached directly on the platforms, which will makes easier not only downward translations but also the rotations, since the attaching points are positioned at a certain distance from the geometric center of the platform. Redundancy will depend on the choice of the number of attached wires. Rotations and translations can be obtained by actuating, as for the previous groups, the wires either symmetrically or asymmetrically.

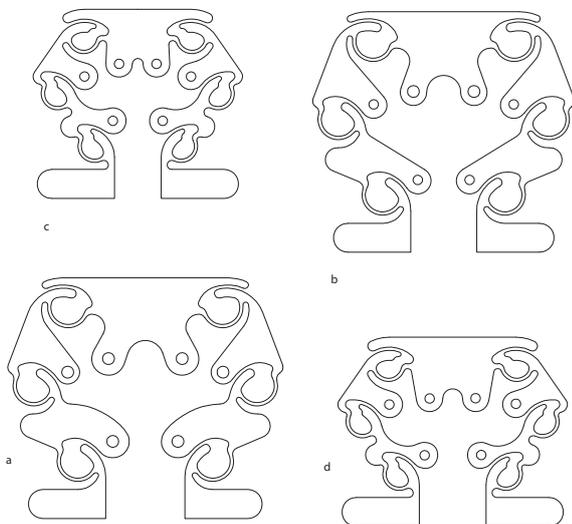


Figure 9 Geometries of 3 D.O.F. planar platforms belonging to the fourth group: from a) to d), models with decreasing overall dimensions and a different wire attaching method.

#### 5 FINITE ELEMENT ANALYSIS ON THE FOUR GROUPS OF STRUCTURES

The structures proposed have been analyzed before starting the phase of construction. The idea of using traditional analysis methods to study compliance in mechanisms has been used many times in the last decades. As for only representative example, in [2] a precision MEMS-based six degrees-of-freedom (DOFs) manipulator has been presented and the analysis has been based mainly using flexible beam theory and FEM modeling.

Some solid models have been built and then prepared for FEA. The constraints have been placed on the bottom of the base links. For the first two groups, the traction force due to the cable tensions has been modeled as a negative pressure acting on the section of the cable. For the other two groups, a traction force has been introduced directly on the pseudo-rigid links in order to simulate the action of the

attached wires. Finally, pressure forces have been added to the surfaces of the conjugate elements that get in contact. The structures have been assumed to be in nylon, while the mesh has been automatically generated by imposing an adequate accuracy in analysis. Loads have been imposed to remain reasonably within the linear behaviour of the material.

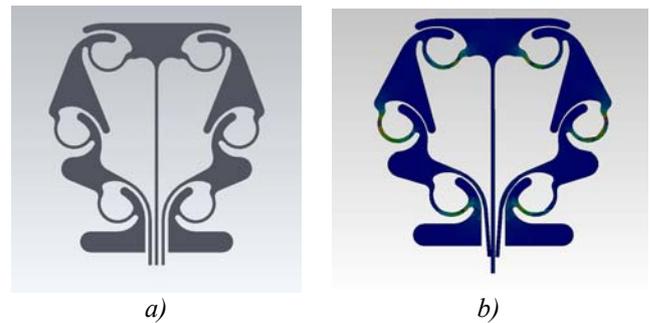


Figure 10 View of the 3D model with 3 cables: a) unloaded structure; b) the structure under the effect of the applied load (treated with FEA).

#### 5.1 FEA OF THE FIRST GROUP

With reference to Figure 10, this platform seems to have the advantage of a reasonably large range of vertical translation. The main drawback is a smaller range of angular variations of the platform orientation.

#### 5.2 FEA OF THE SECOND GROUP

With reference to Figure 11, this platform seems to behave very similarly to the one presented in group 1. The platform vertical translation capacity is similar, while the rotation attitude is a bit greater but still quite limited.

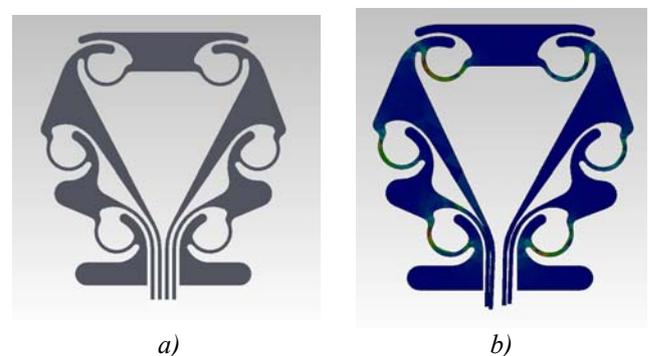


Figure 11 View of the 3D model with 4 cables: a) unloaded structure; b) the structure under the effect of the applied load (treated with FEA).

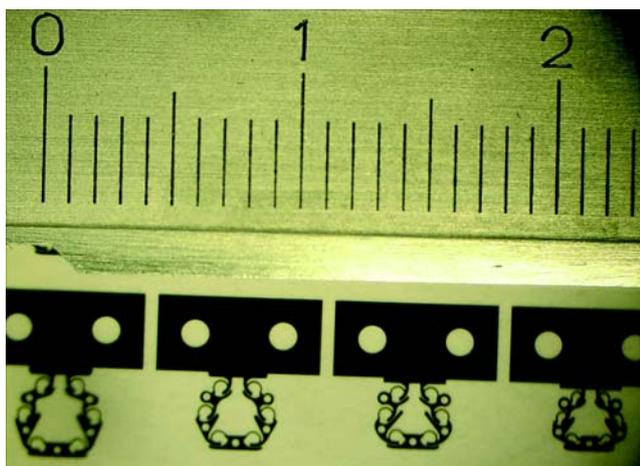


Figure 14 Comparison between a ruler and a group of microrobot projections on the photographic film.

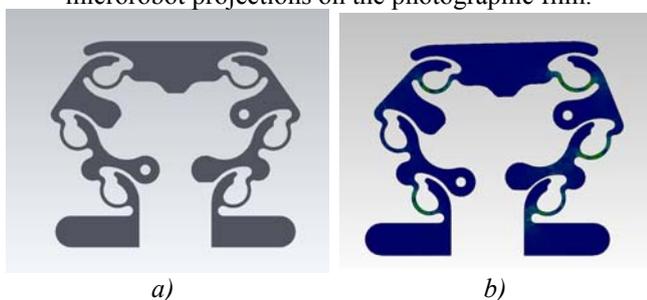


Figure 12 View of the 3D model with different way of wire attaching: a) unloaded structure; b) the structure under the effect of the applied load (treated with FEA).



Figure 13 View of the 3D model with different way of wire attaching: a) unloaded structure; b) the structure under the effect of the applied load (treated with FEA).

### 5.3 FEA OF THE THIRD GROUP

With reference to Figure 12, this platform seems to have behaviour similar to the one presented in group 2. The platform vertical translation capacity is similar, while the rotation attitude is a bit greater.

### 5.4 FEA OF THE FOURTH GROUP

With reference to Figure 13, this platform seems very promising, because FEA showed that the platform vertical translation capacity and the rotation attitude are greater than in the previous cases.

## 6 PROTOTYPES FABRICATION

A group of prototypes has been built by using standard microelectronic lithography technique. Figure 14 represents a picture of the photographic film which has been used to obtain different 3D molds on porous silicon with the use of positive photoresist. The actual dimension of the imprinted areas are compared to a ruler with one mark per mm.

After a standard lithographical step, two ways to realize the device have been followed. One consists of the processing of the silicon wafer in plasma etching, which forms the stamp for the following polymer molding. The other one consists of the procedure reported in [5] and [6].

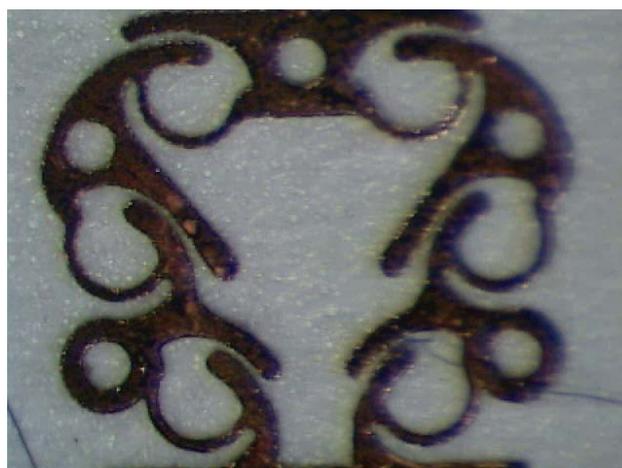


Figure 15 Prototype obtained by using copper electrodeposition.

For the sake of completeness, it is worth noticing that the research group is testing other possible construction techniques. For example, another way to employ the silicon mold consists in its use in metal electrodeposition. According to this technique some prototypes were obtained by using copper electrodeposition, one of which is presented in Figures 15. Finally, by using RIE (Reactive Ion Etching) technology on a 50  $\mu\text{m}$  silicon wafer, some prototypes were obtained, as depicted in Figures 16 and 17.



Figure 17 Silicon prototype (50 μm thick).



Figure 16 Silicon prototype (50 μm thick).

## 7 CONCLUSIONS

This paper has presented an attempt to enlarge the horizons in the realm of MEMS/NEMS-based multi D.O.F. Micro Robots. A new flexural hinge is presented which allows the construction of multi D.O.F. micro robot with planar technique with no need of cumbersome assembly procedures. Non-linear FEA has shown that this solution is feasible for the sake of the required function. Finally, it has been proved experimentally that there are several construction techniques which can be employed for the specific applications and some prototypes have been presented.

## REFERENCES

- [1] Luharuka R. and Hesketh P. J., Design of fully compliant, in-plane rotary, bistable micromechanisms for MEMS applications. *Sensors and Actuators, A* 134 (2007) 231238.
- [2] Brouwera D. M., De Jongc B. R. and Soemers H. M. J. R., Design and modeling of a six DOFs MEMS-based precision manipulator. *Precision Engineering*, 34 (2010) 307319.
- [3] Kota S., Joo J., Li Z., Rodgers S. M. and Sniegowski J., Design of compliant mechanisms: applications to MEMS. *Analog Integrated Circuits and Signal Processing*, 29, 715, 2001.
- [4] Jang G.-W., Kim K. J. and Kim Y. Y., Integrated topology and shape optimization software for compliant MEMS mechanism design. *Advances in Engineering Software*, 39 (2008) 114.
- [5] Balucani M., Conductive microstructure obtained by converting porous silicon into porous metal, European Patent Application EP20080425186.
- [6] Balucani M., System for contacting electronic devices and production processes thereof, United States Patent Application 20080012114.
- [7] Belfiore N.P., Scaccia M., Ianniello F., Presta M., Selective compliance hinge, World Intellectual Property Organization, WO/2009/034551, Int. Appl. No. PCT/IB2008/053697, Publ. Date March, 19th, 2009.
- [8] Belfiore N.P., Scaccia M., Ianniello F., Presta M., Perfetti L., Selective compliance wire actuated mobile platform, particularly for endoscopy surgical devices, World Intellectual Property Organization, WO/2009/034552, Int. Appl. No. PCT/IB2008/053698, Publ. Date March, 19th, 2009.

# SOME ISSUES ON HOLONIC SYSTEMS ANALYSIS, DESIGN AND IMPLEMENTATION

Doru Panescu    Carlos Pascal

“Gheorghe Asachi” Technical University of Iasi,  
Department of Automatic Control and Applied Informatics, Iasi, Romania

## ABSTRACT

This contribution reveals some problems concerning the use of holonic manufacturing systems. These regard both the design and implementation phases and they can be discovered since the system analysis. A new Petri net based model is proposed for the holonic system organized according to the Contract Net Protocol coordination scheme. Certain weaknesses are discovered both at the model and implementation levels and some solutions are sketched. These are illustrated with a case from an experimental manufacturing environment that could exemplify the problems regarding the resource sharing and holons' coordination.

Keywords: Holonic Manufacturing Systems, Petri nets, Multiagent Systems, Contract Net Protocol

## 1 INTRODUCTION

To face the present requirements of manufacturing environments some control architectures have been proposed, like the intelligent control schemes, holonic and multiagent systems. One problem with these approaches is the lack of a theoretical background that should guide the analysis, design and implementation phases, they being often conducted in a heuristic manner. The present contribution aims at identifying some problems that can occur when ad-hoc solutions are used to carry out the manufacturing planning and controlling methods, and at proposing certain solutions for solving them.

Within the range of manufacturing control schemes, the Holonic Manufacturing Systems (HMSs) have a privileged position. Among their advantages, the adaptive, autonomous operation, the optimal resource use and real-time response to fast demands and environment changes are to be mentioned. All these stem from a specific combination between hierarchical and heterarchical control approaches [1], the way an HMS is able to reconfigure itself so that the available resources can be optimally used and the asynchronous, continuously received manufacturing goals

can be in-time solved. PROSA is a reference holonic architecture, which is frequently used and we considered, too [1], [2].

PROSA holonic scheme identifies the types of entities (holons) and specifies their main connections. The holons' relations rise new subjects to think about, regarding the coordination protocol, the way planning and execution have to be mixed, the specific constraints to be considered in various phases of system deployment so that certain HMS performance can be guaranteed. These aspects are discussed in the present contribution, which is organized as follows. First, the HMS approach is briefly presented and the problems to be tackled when using it are commented. A Petri Net (PN) model for a generic holon is proposed, in two stages: first a simplified, high level version, then a more detailed one. This is followed by some information regarding the implementation, which complete the good operation conditions discussed in the previous section and give some illustrative examples regarding an experimental manufacturing system. Finally, a few conclusions are derived.

## 2 MAIN ISSUES OF A HOLONIC SCHEME

A few guideline points for HMS design and implementation have been already established [2], [3], [4], [5]. In the proposed approach, each holon is a composite entity, possibly distributed, incorporating a software agent as its deliberative part and a structural component as an

---

Contact author: Doru Panescu

Address: Str. Prof. dr. D Mangeron 27, IS 700050, RO

Phone: + 40 232 230751

E-mail: dorup@tuiasi.ro

operational part [3]. Thus, the aggregation process required in the holonic system takes place within the structural component, which can be either a physical device – like it often happens with the resource holons, or a holarchy formed with other holons, which is usual for an order or a product holon.

One can adopt some coordination principles from multiagent systems, as the HMS should involve several agents interacting during the manufacturing cycle. Meanwhile, certain differences exist between a multiagent system and an HMS [4], [5]. In the holonic approach, the agents' construction and communication must be appropriately adapted. The main points of the considered solution are:

- Each holon contains an agent that has to solve the planning phase and guide the control actions.
- The agent reasoning mechanism is the BDI scheme [6], which provides certain benefits regarding the adaptive, fault-tolerant operation.
- The coordination procedure is the Contract Net Protocol (CNP), commonly used in agent based systems, which was modified and enhanced to fit the HMS requirements [4], [7], [8].

A central issue for HMS regards the holarchy formation. This is a goal driven process: as soon as a new manufacturing order appears and must be solved, a holarchy should be organized. It will comprise one or more order holons, depending on the relation established between a production order and the goals to be solved by the HMS. A common case is when each order is related to one or more manufacturing goals, and the proposed scheme considers an order holon to be in charge with solving each goal. Then, starting from this order holon the corresponding holarchy must be organized (see Fig. 1). The order holon launches a certain number of goals (sub- goals of the originating goal) that will represent various production activities. These are taken over and solved by one or more product holons, those possessing knowledge on the respective production operations. The product holons will further apply for the resource holons' cooperation, namely to carry out the activities related to each production function. The resource holons' operation is still a goal driven one, as they are activated by the tasks to be solved decided and announced by the product holons.

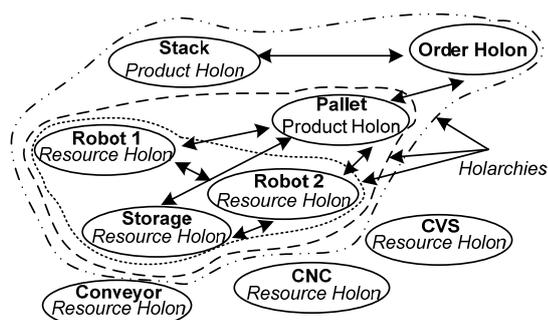


Figure 1 A holarchy determined by an order holon regarding a single pallet.

Furthermore, a resource holon can itself launch a goal when it is able to solve only part of a task issued by a product holon and it asks the cooperation of other resource holons.

The scenario corresponding to the holarchy organization of Fig. 1 regards a manufacturing environment considered for experiments [9], which contains two industrial robots, a conveyor, a computer vision system, a machine tool and several storage devices. The manufacturing orders refer to obtaining distinct pallets or stacks of pallets, so that two product holons are implied. One is able to manage the operations regarding the achievement of pallet stacks, while the other knows how a specified pallet layout can be attained. Namely, until 16 parts can be placed on a pallet, and the pallet arrangement can differ depending on the parts' positions and the parts' types (the parts are of 5 types). Fig. 1 illustrates a holarchy (that will be further considered) obtained when a manufacturing order for obtaining a single pallet with a specified layout is handled by an order holon. One can see how a holarchy can contain sub-holarchies.

It is clear that for the manufacturing order processing in an HMS the CNP can be the right strategy to be involved: a goal provider holon (in fact its agent as the decisional component) acts as a manager, while the holons that can solve the goal or part of it are making bids, meaning they act as contractors according to the CNP coordination scheme. Thus, the main principle for the holarchies' formation is established: each time a manufacturing order appears, this is split in goals and sub-goals until the elementary manufacturing operations are identified and ready to be done by the resource holons. Though the mechanism seems to be simple, some issues have to be further clarified:

- How several holarchies are formed and operate simultaneously when the HMS has to face several manufacturing orders. In fact, even a single order can impose a large number of goals/sub-goals and the coordination mechanism must be properly designed.
- Taking into account the above mentioned point, one has to consider the possibility of deadlocks and infinite loops and find the means to avoid them.
- The optimal solution is not always guaranteed at the CNP application [7], so that the holarchy formation must be based on certain specifications that should lead the operation towards the optimal results for the HMS.

The next sections of this contribution provide some answers to these problems.

### 3 A MODEL FOR HOLONIC COORDINATION

To surpass the problems of HMS design and implementation a specific modelling and analyzing methodology is needed. With respect to this, the PN model has been sometimes used [10], [11], [12] but one can remark that some open problems are still present. Thus, the strategy described in [10] does not clarify the entire HMS organization, but it mainly takes into account the relation

between a product holon and some resource holons that are making bids for the respective product holon; meanwhile the relation between the planning and execution stages is not fully specified. The model proposed in [12] does not make the connection with the planning phase, and the relation with the CNP is only briefly treated. In a real HMS, one or more order holons must be considered and several links are established between these holons and the product/resource holons implied in solving a manufacturing order. Nonetheless, the case when the same kind of holons are implied in a hierarchical relation has to be tackled, as the situation when a resource holon needs the cooperation of other resource holons (for example, in our manufacturing environment a robot resource holon can send goals to be solved by the computer vision system, the storage device holons or by other robots). Then, the bids' transmission and their selection by the manager have to be treated in all the cases, namely when there is one or more offers for the same goal, and these are sent/received asynchronously. The modelling method must support the case when several holarchies simultaneously co-exist and their safe operation must be determined. All these problems are not handled or only partially solved by the HMS models so far.

The model proposed in this paper takes into account the above mentioned problems and the necessity to combine the CNP as the problem decomposing mechanism with the BDI reasoning procedure that has its role in plans' selection and solution finding. Moreover, an agent based programming implementation allows the interleaving between planning and execution. The first level PN model for an agent that operates as the holon's deliberative component is depicted in Fig. 2.

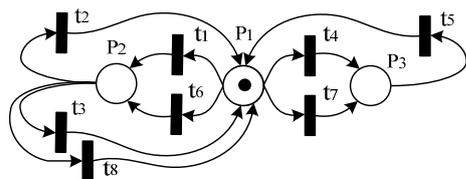


Figure 2 A PN model for the basic holon activities.

While the common PN notation is used [13], various places represent the agent's states and the transitions embody the events. The significance of the three places is:  $P_1$  is the agent idle state,  $P_2$  is the agent's planning state and  $P_3$  represents the agent's execution phase. The considered transitions are:  $t_1$  represents an event of receiving a goal,  $t_2$  appears when the agent sends goals to be solved by other holons (this happens when it acts as a manager),  $t_3$  regards the event of bids' sending (this refers to a moment when the agent acts as a contractor),  $t_4$  represents the event when the agent receives a contract that was awarded as a consequence of a previous bid or the message that its bid was not selected,  $t_5$  represents the event of a control/execution operation end,  $t_6$  represents the event of a bid reception as a consequence of a previously issued goal,

$t_7$  refers to an event launched when the execution phase has to be resumed, and  $t_8$  regards an event issued from the planning stage when the holon acting as a manager notifies the unselected contractors so that they should be able to deallocate their resources.

At the first decomposition level this model reflects the agent activity phases within the holon operation. Thus, an agent enters the planning phase as soon as it receives a goal to solve (transition  $t_1$ ). First, it tries to solve the goal by itself acting as a contractor, and two follow-ups are possible in an enhanced form of the CNP [4], [8]. It finds a solution and continues with sending a bid (transition  $t_3$ ) or it cannot find any solution for the respective goal and again sends a bid, representing the negative answer (this event is represented by  $t_3$  too, because only the parameters being transmitted are modified). The second continuation regards the case when the agent can solve only part of the goal, which makes it to send some goals/sub-goals to the other holons (transition  $t_2$ ); depending on the received bids (transition  $t_6$ ) it will be able to continue with one of the above mentioned transitions, by passing through the planning phase (place  $P_2$ ).

The other agent's stage, the execution one, appears after a contract awarding (transition  $t_4$ ) and it will be followed by a transition to the idle state when a control sequence is ended. This is to be understood as the agent is only sending the main control parameters and commands towards the holon structural component. For example, in the case of the resource holons this structural component may be a robot controller, a PLC controlling a machine tool, etc. It is the role of the structural part to carry out a specific control algorithm. For the order and product holons, the execution phase is composed of activities regarding messages' submission toward the agents representing contractors that have to carry out the decided plans. This analysis shows that in all the cases a single continuation is possible from the holon execution, namely the idle state. One has to understand that the way an execution is ended in the manufacturing environment with success or failure should be reflected by an appropriate message sent to the agent (an event), modelled by the transition  $t_5$  regarding the appearance of a new goal.

It may be the case that a plan execution implies a sequence of actions. After launching an action the agent passes to the idle state (by transition  $t_5$ ), enabling the consideration of further activities (for example, a planning phase regarding another goal). When the necessity to continue the execution appears, the event modelled by the transition  $t_7$  will lead again the agent to the execution stage. These explanations show how our model is able to represent the planning and execution interleaving, as an important feature of the agent based HMS operation.

In our model a special meaning has the idle state. It is the only situation when the agent can take into consideration a change in its environment. Both events regarding modifications in the manufacturing process, as detected by the sensorial systems of the resource holons (various

sensors can be included into the structural component of a resource holon), and also events regarding the other agents, as their messages that have to be received (e.g. a new goal that must be processed) can be taken into account in the idle state. This means that a holon is able to monitor its environment in a quasi-continuous manner. When a change appears, after being detected by a sensorial component, it is considered by the deliberative part (the agent) only if this is already in the idle state, or after finishing the on-going planning or execution phase.

The model of Fig. 2 has to be properly understood regarding the way a holon can process several goals in the same time; these will be in different stages of their evolution. The initial marking of the PN is with one token in the place  $P_1$  and no tokens in the other places (first the holon's agent is in the idle state). The agent's activity is started when the first goal is received, this leading to a transition to the planning state. If after that moment one or more goals appear, they will be judged in sequence only when the agent is freed from a planning or execution stage, being again in the idle state. In fact, the goals asynchronously sent to the agent are kept in a queue. Even so, the model of Fig. 2 is correct, as the head goal in the queue will launch the transition  $t_1$ , as soon as the agent has paused or finished the operations on a previously received goal. This multi-goal operation of its agent makes the holon analysis more difficult and some further explanations can be given by considering a more detailed model.

Fig. 3 is a refinement of the PN modelling the agent operation within a holon. Again the general case is considered, namely the respective PN is valid for all the three holons' types, with a few differences for the order holons that will be finally identified. In this model the places represent both states and preconditions of certain events. Thus, when a goal has been received (the transition  $t_1$  represents the submission of a goal from the exterior of the holon, because we can also discuss about internal issued goals, namely the transition  $t_8$ ), it will be processed when the agent is in the idle state (place  $P_0$ ). Now the model can take into account the receiving of several goals – the transition  $t_1$  produces tokens for the place  $P_1$ . According to the BDI reasoning mechanism

a further condition for a goal treatment is the existence of at least one plan to be related with the respective goal, this requirement being modelled by the place  $P_2$  (the BDI mechanism is providing tokens in the place  $P_2$ ). Fig. 3 shows the three possible cases regarding the goal processing:

- if the agent has a plan that entirely solves the goal, the transition  $t_2$  will be fired;
- if the agent's plan can only partially solve the goal, the transition  $t_9$  is activated; in this situation the agent, which is contractor for the current goal, becomes also manager for the part of the goal it cannot solve and the part of the PN starting with  $P_8$  models this further application of the CNP;
- if the agent has no solution for the present goal a return to the idle state follows, started by the transition  $t_{16}$ ; namely, the state represented by the place  $P_4$  is determining a negative answer (a bid with a negative cost) that is sent towards the manager originating the goal (the transition  $t_{17}$ ); in the BDI mechanism even in this case a plan must exist, which explains why  $t_{16}$  has  $P_2$  between its input places.

The place  $P_3$  regarding the state when the agent has found a complete solution for a goal is linked with the transition  $t_3$  representing the event of sending a bid towards the manager that issued the respective goal. The idle state that follows this transition is the unique situation that represents the agent waiting for an event that will launch a new agent's processing activity. We preferred to repeatedly draw this place ( $P_0$ ) in the PN to make it more comprehensible. The transition  $t_5$  nicely represents how the holon execution is started (the execution state is modelled by the place  $P_6$ ) after a whole solution was found for a goal, only if two conditions hold: the agent is in the idle state and it was awarded the contract after the firing of transition  $t_4$  (as with  $t_1$ , this transition has no input place because it regards an event that is not originated from an agent's state, but from the exterior of the agent). Another possibility to resume execution is when an event regarding the necessity to continue an already approved plan with its further actions appears (transition  $t_{19}$ ). If the agent receives a message announcing it that was not selected (transition  $t_{20}$ ) it

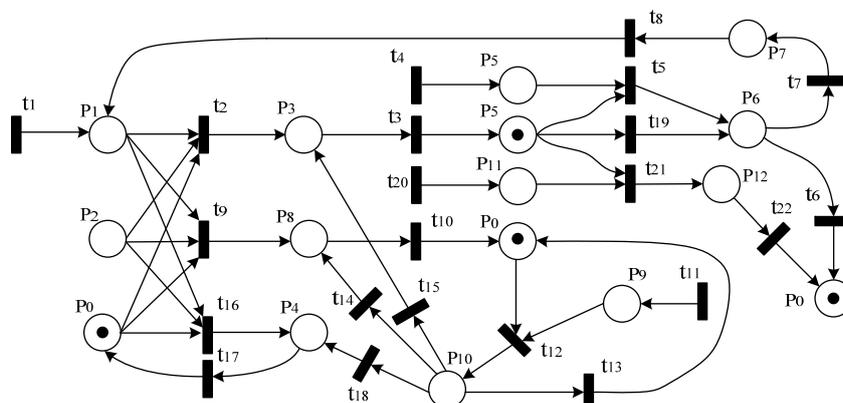


Figure 3 A detailed PN model for a holon applying the CNP coordination mechanism.

continues with a state of resource deallocation (place  $P_{12}$ ). Now, this extended PN reveals the two cases regarding the execution ending. If this is finished successfully the transition  $t_6$  leads towards the idle state. When the execution fails, the transition  $t_7$  conducts the agent towards a state allowing the generation of internal goals (this event is represented by the transition  $t_8$ ). The treatment of these newly generated goals should eliminate/surpass the consequences of the execution failure.

As already mentioned, the part of the PN starting with the place  $P_8$  regards the situation when the agent has to play both the manager and the contractor role in the CNP. The agent is trying to complete a bid and it sends certain goals to be solved by other holons (transition  $t_{10}$ ). An externally generated event (transition  $t_{11}$ ) representing a bid reception can conduct towards the state when the agent acts as manager and is selecting the bids (the place  $P_{10}$ ). The continuation from  $P_{10}$  is dependent on whether a complete solution was found or not, through the transitions  $t_{15}$  and  $t_{14}$ , respectively. These transitions determine also the messages sent to the agent's contractors that were not selected so that they should deallocate their resources and be able to participate in solving other goals.

As already remarked, the PN shown in Fig. 3 can represent all the types of holons. A few changes appear for an order holon that is the first to receive a manufacturing order. The corresponding agent would never be a contractor, but only a manager. Thus, the respective PN does not contain the places  $P_5$ ,  $P_{11}$ ,  $P_{12}$  and the transitions  $t_4$ ,  $t_5$ ,  $t_{20}$ ,  $t_{21}$ , and  $t_{22}$ , while the output place of transition  $t_3$  becomes  $P_6$ .

The developed PN can be used to analyze the holon's performance. A correct holon operation is obtained if the marking with a token in the place  $P_6$  is reachable from the initial marking (as drawn in Fig. 3, the initial marking is characterized by a token in  $P_0$  and no tokens in the other places). One can notice that the respective reachability condition holds if the agent is not kept in one of the three possible loops:  $P_0 t_{16} P_4 t_{17} P_0$ ,  $P_8 t_{10} P_0 t_{12} P_{10} t_{14} P_8$  and  $P_0 t_{12} P_{10} t_{13} P_0$ . These loops correctly reflect the wrong operation situations regarding: goals that were not provided with solving plans, sub-goals that indefinitely do not receive a complete solution as only incomplete bids are made, and respectively sub-goals that cannot be finalized due to the lack of bids, for example when some resources are missing. Certain real cases that reflect such circumstances in our manufacturing environment are commented in the next section, and some solutions are provided, too. It is to mention that the developed PN model may enable other characteristics of the HMSs to be analyzed.

#### 4 IMPLEMENTATION ISSUES AND EXPERIMENTS

The holonic concept finds an efficient implementation over the multi-agent platforms [5], [14]. In our design and implementation (further details can be found in [3], [4], [8]), the JACK agent intelligent platform is used [15]; it offers

the needed support to apply the holonic method. This section completes the already discussed issues regarding the HMS operation with some conditions to be considered at the agents' implementation. These were established in order to avoid the problems being identified in the modelling phase and in the testing experiments.

In the planning process, a holon that plays the manager role (e.g., an order or a product holon) can determine an operation or a set of operations for an identified or non-identified object. In the considered manufacturing environment such an object can be a raw part or a semi-product that will be involved in a specified operation to form the needed product. The operation can refer to a transport, processing, storing or an assembling action. These activities are carried out by the resource holons. No matter the object is identified or not, because sometimes the same operation can be fulfilled by more resources, some drawbacks may appear (like the loops mentioned in the previous section) conducting to certain conditions to be imposed on the product and resource holons.

Let us consider the two situations regarding the use in the planning phase of identified or non-identified objects. When the object is designated by an identifier and a piece of information about its situation (as with the holon owning an object), a problem can still appear on the optimal solution. It is possible that several objects with certain identical properties exist, but only one of them can lead to the optimal solution. It happens that the product holon which is the manager does not possess knowledge to choose the right object. To avoid this, the product holon must consider the whole set of objects, and propose the same goal for each of them; after that, it can be able to select the optimal bid, as transmitted by the resource holons that should have more precise information on each action's cost.

In the second case, the product holons propose goals without indicating a specific object. This time the resource holons, those that can accomplish the goal, will have to select the right object. For this situation, in certain conditions the optimal solution may be not reached, or even no solution obtained. To easily understand the problems that can appear, let us consider the actual case of the storage device resource holons. A goal with a non-identified object for the respective holons can refer to finding an object with certain characteristics, or a free position to be filled in. When the resource holon makes an offer, the object or location is locked. Thus the respective entity is no longer available for another bid. Only if the offer is not used in the manager's planning process, the implied entity is unlocked.

This behaviour of the storage device resource holon can limit the solution space when it could make bids to several managers, and also when such manager holons can propose the same goal. Furthermore, the storage holon cannot make a difference between more goals and/or sub-goals referring to the same operation. An illustrative example from our manufacturing environment is the one sketched in Fig. 4. Here the goal initiated by the product holon named

*Pallet\_PH* (see the holarchy of Fig. 1, too) is to move a black, non-identified object into the indicated rectangle location on the storage device. The arrow marked with *c* indicates that the final position is locked by the holon *Pallet\_PH*. For this goal, two robots try to find a black object in their working area (its presence and location are not identified), asking the cooperation of the storage holon (the storage device is placed within an area that is accessible to both robots). If this sends a single bid to one of the two robots, then the other robot cannot make a bid for the respective goal, even if the cost that it is able to offer would be the best (e.g., its handling and part moving operations can be done in a shorter time in comparison with the other robot).

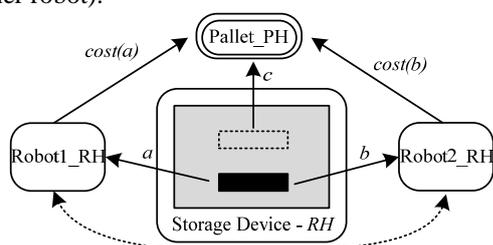


Figure 4 A scenario illustrating the problems for a collaborative solution.

The optimal solution can be obtained if the *Pallet\_PH* holon receives bids from both robots' holons. To make this possible, as well as to solve the other above mentioned issues, some conditions must be fulfilled:

- the operations defined for the non-identified objects must be labelled by the product holons with a unique default object identifier; the holons that receive goals can make the difference between the default and real object identifiers based on a specific flag, or the object's location that is stated or not;
- every sub-goal issued by the resource holons must hold the object's identifier; for the non-identified objects,

the location must remain blank if the holon is not the one that holds the respective object;

- when a resource holon that holds a needed object makes an offer it must assert the object identifier to the real one; moreover, when more goals with the same mark arrive at the same holon from different managers, they will get the same object in their bids.

Using these conditions, for the above example (see Fig. 4) both robots receive an identical bid from the storage device resource holon (regarding the available part), and both can calculate the cost for the bid they are supposed to make to the manager holon (the pallet product holon). Only thus our HMS can reach the optimal solution.

Another kind of drawback was detected by simulated and real-life conducted experiments. This is about possible infinite loops or partial deadlocks. They appeared when the same type of resource holons acted as managers for a common goal. It is to remark that such a situation is already highlighted by the proposed PN model, too. It regards the loop that can appear between the places  $P_8 P_0 P_{10} P_8$  when there is no received bid through the transition  $t_{11}$ . Considering the example presented in Fig. 4, if there is no black part for the current goal in the manufacturing environment, both robots will try to collaborate hoping that when acting as managers they can receive assistance from the other robot to solve the goal. This infinite loop is represented by the dotted arrow between robot holons in Fig. 4. In this case, the solution to avoid the loop is to make use of the above presented conditions, so that each holon having the manager function be able to notice this kind of loop. The detection is possible by using the objects' marks.

The agents' holons interaction diagram for the above example, when using the mentioned conditions, is captured in Fig. 5. The product holon *Pallet\_PH* has two goals. The first goal is to reserve a final position for a black part (this is marked in the diagram of Fig. 5 as being of type E), and after that to transfer the part in the position that was found

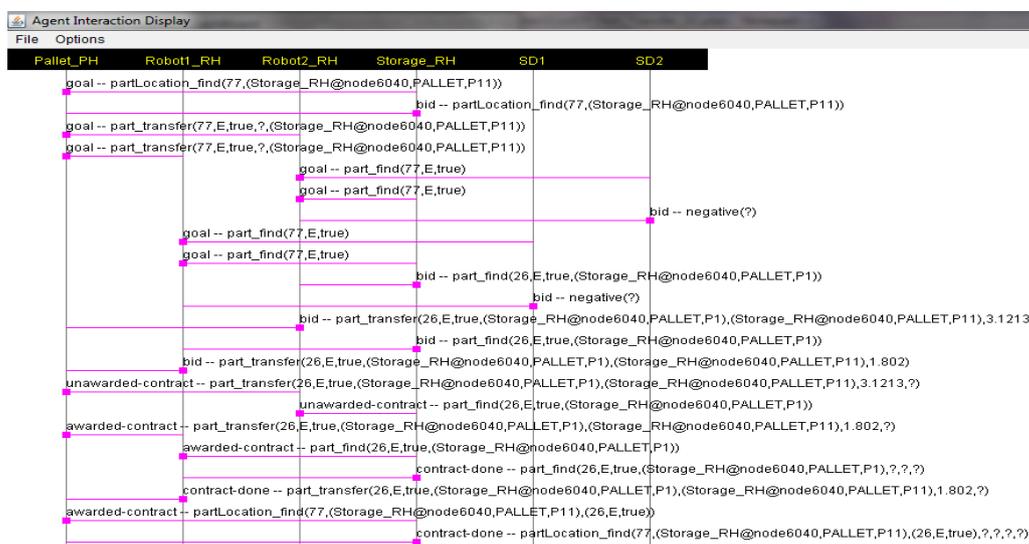


Figure 5 The inter-agent holon communication for the planning and execution phases.

(position *P11* in the diagram). The holon *Pallet\_PH* marks the needed part with an identifier (the label 77 in the diagram). For the second goal, the robot resource holons obtain the same bid from the single storage device holon (*Storage\_RH*) which includes the real object's identifier (this is the label 26). In the respective experiment the holon *Pallet\_PH* has chosen the bid of the holon *Robot1\_RH* as being the best, thus finalizing the goal.

Fig. 5 contains the agents' message exchange for the execution phase, too. One can see how the manager holon sends messages on contract un-awarding, and how the holons that received the contracts carry out their actions and finally send confirmation messages. It is to notice that for the execution phase, an awarded contract determines the updating of the necessary information; for example, in our case the storage device holon updates its knowledge base regarding the release of a location and a new object that was stored.

## 5 CONCLUSIONS

By combining a theoretical sound model with systematically conducted design and implementation phases the HMS approach can become a real solution for the industrial control systems. In the presented methodology the difficulties regarding the way the decisional and control mechanisms are distributed over several holons could be surpassed by complying to a certain deployment cycle: an adequate model is derived, then the constraints that the model imposes for a safe and optimal operation are taken into account, and finally these are adapted and extended as needed at the design and implementation stages. Thus, the originality of this contribution regards both the new PN model that addresses all the types of holons and reflects the holons activities and the holarchies' formation, together with the way this model is applied by considering specific implementation mechanisms for the holons' agents.

The already made experiments, which have only briefly presented in this contribution, show two conclusions. First, specific constraints must be taken into account so that the HMS should be able to find a solution to any kind of manufacturing goal and then so that this to be the optimal one. Secondly, it is clear that an HMS determines better results in comparison with a classical approach (an improved adaptability to new and fast changing orders, a shorter manufacturing time, and a better resources' use). Further experiments are planned for well chosen scenarios that could allow a clear comparison with a classical control scheme, and an improvement of the PN model is expected, too (the consideration of the coloured PN is supposed to better model the HMS processing of several goals).

## REFERENCES

[1] Van Brussel H., Wyns J., Valckenaers P., Bongaerts L. and Peeters P., Reference architecture for holonic

- manufacturing systems: PROSA. *Computers in Industry*, Vol. 37, No. 3, pp. 255–274, 1998.
- [2] Valckenaers P., Van Brussel H., Wyns J., Bongaerts L. and Peeters P., Designing holonic manufacturing systems. *Robotics and CIM*, 14, pp. 455 – 464, 1998.
- [3] Panescu D., Sutu M. and Pascal C., On the design and implementation of holonic manufacturing systems. *Proc. of the CSIE 2009*, Los Angeles/Anaheim, USA, pp. 456 – 461, 2009.
- [4] Panescu D., Varvara G., Pascal C. and Sutu M., On the design and implementation of the resource holons in a PROSA based Architecture. *Proc. of INES 2009*, Barbados, pp. 101 – 106, 2009.
- [5] Leitao P., Agent-based distributed manufacturing control: A state-of-the-art survey. *Engineering Applications of Artificial Intelligence*, Vol. 22, No 7, pp. 979-991, 2009.
- [6] Wooldridge M., Intelligent Agents, in: *Multiagent Systems. A Modern Approach to Distributed Artificial Intelligence* (G. Weiss, Edit.). MIT Press, Cambridge, pp. 54 – 60, 2001.
- [7] Huhns M. and Stephens L., Multiagent Systems and Societies of Agents, in: *Multiagent Systems. A Modern Approach to Distributed Artificial Intelligence* (G. Weiss, Edit.). MIT Press, Cambridge, pp. 100 – 103, 2001.
- [8] Panescu D., Pascal C., Sutu M. and Varvara G., Collaborative robotic system obtained by combining planning and holonic architecture. *Proc. of AT-EQUAL*, Iasi, pp. 138-143, 2009.
- [9] Panescu D. and Varvara G., On the sensorial integration for a holonic manufacturing system. *Proc. of the 3rd UKSim European Modelling Symposium on Computer Modeling and Simulation 2009 (EMS 2009)*, Athena, pp. 367 – 372, November 25-27, 2009.
- [10] Hsieh F-S., Holarchy formation and optimization in holonic manufacturing systems with contract net. *Automatica*, 44, pp. 959 – 970, 2008.
- [11] Hsieh F-S., Analysis of contract net in multi-agent systems. *Automatica*, 42, pp. 733 – 740, 2006.
- [12] Leitao P., Colombo A. and Restivo F., An approach to the formal specification of holonic control systems. *Holonic and Multi-Agent Systems for Manufacturing, Lecture Notes in Artificial Intelligence*, Springer-Verlag, pp. 59-70, 2003.
- [13] Murati T., Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4), 1989, pp. 541 -580.
- [14] Bordini R., Dastani M., Dix J. and Seghrouchni A. E. F. (Eds.), *Multi-Agent Programming. Languages, Platforms and Applications*. Springer, 2005.
- [15] \*\*\* *JACK™ Intelligent Agents – Agent Manual*, Release 5.3, Agent Oriented Software Pty Ltd, Carlton South, Australia, 2005



# OPTIMIZING PARAMETERS OF TRAJECTORY REPRESENTATION FOR MOVEMENT GENERALIZATION: ROBOTIC THROWING

A. Gams, T. Petrič, L. Žlajpah, A. Ude

Department for Automation, Biocybernetics and Robotics  
Jozef Stefan Institute, Ljubljana, Slovenia

## ABSTRACT

For effective use of learning by imitation with a robot, it is necessary that the robot can adapt to the current state of the external world. This paper describes an optimization approach that enables the generation of a new motion trajectory, which accomplishes the task in a given situation, based on a library of example movements. New movements are generated by applying statistical methods, where the current state of the world is utilized as query into the library. Dynamic movement primitives are employed as the underlying motor representation. The main contribution of this paper is the optimization of dynamic movement primitives with respect to the kernel function positions and over the entire set of demonstrated movements. We applied the algorithm to a robotic throwing task, where the location of the target is determined by a stereo vision system, which can detect infrared markers. The vision system uses two Nintendo Wiimotes for cameras.

Keywords: generalization, optimization, dynamic movement primitives

## 1 INTRODUCTION

One of the issues in modern robotics is the problem of trajectory generation and modulation [4]. Manipulation of objects at arbitrary locations in the 3D world requires the robot to either learn all the movements, which is difficult as the learning space to explore is huge; to modify the previously learned trajectories; or to generalize from existing knowledge [13, 5, 15].

Different approaches have been utilized for modulating learned trajectories. Different approaches include also different representations of the trajectory. The representation of the trajectory is crucial for an effective algorithm [6]. Among the simplest is the storing of large time-indexed vectors [10]. Alternatively, more compact representations can be constructed by using interpolation algorithms such as spline fitting or storing only via-points

[13]. Hidden Markov Models (HMM) are one of the possibilities [2], which can be effectively utilized for the modulation and also generalization. Another approach is the use of Gaussian mixture models [3].

One of the approaches is to utilize nonlinear dynamic systems as a basic motor representation. The approach utilizes a set of nonlinear differential equations, which can approximate arbitrary trajectories, and are termed *dynamic movement primitives* or DMPs [9, 8]. Researchers have shown that by changing the underlying differential equations, dynamic movement primitives can also be modified in several ways to modify the approximated trajectories, e.g. as in [14]. These modifications have to be implemented by an expert, and cannot be utilized autonomously.

Generalization from a library of learned movements, on the other hand, can autonomously generate new trajectories to account for differences in the external world. Different approaches exist in the generalization, for example one can use HMMs [2] or DMPs [5]. In our approach we use the DMP representation. The parameters of DMPs include the number of kernel functions, their distribution over the course of the movement, and their width. It is common to use a linear distribution over the course of the trajectory, and kernel functions of uniform width.

---

Contact author: A. Gams

Jozef Stefan Institute  
Department for Automation, Biocybernetics and Robotics  
Jamova cesta 39, 1000 Ljubljana, Slovenia  
E-mail: andrej.gams@ijs.si

The main purpose of this paper is twofold. Firstly, we want to experimentally evaluate a methodology for generalizing example trajectories on a dynamic task with explicitly non-intuitive connection between the final movement position and the query point. An example of such task is robotic throwing, where the final position of the robotic arm, and the landing spot of the projectile are not directly related. Secondly, and this is the emphasis of the paper, we want to show that we can optimize the used dynamic movement primitives representation in kernel function locations and widths to achieve better trajectory approximation.

In the next sections we first introduce the DMPs in Section II. We then explain the generalization algorithm and the optimization of kernel function locations and width in section III. In Section IV we present a demonstration task of throwing a ball with a robot to a target position. 2 Nintendo Wiimotes are used for a stereo vision system to determine the location of the target. A summary is given in section V.

## 2 DYNAMIC MOVEMENT PRIMITIVES

In this section we give a brief overview of the DMPs, which were first introduced in [9, 8].

For a single degree of freedom denoted by  $y$ , which can either be one of the internal joint angles or one of the external task space coordinates, the following system of linear differential equations with constant coefficients has been proposed as a basis for motion specification

$$\tau \dot{z} = \alpha_z (\beta_z (g - y) - z), \quad (1)$$

$$\tau \dot{y} = z. \quad (2)$$

The parameters  $\alpha_z, \beta_z$  and  $\tau$  are positive constants and with  $\alpha_z = 4\beta_z$  this system has a unique attractor point at  $y = g, z = 0$ .

Equations (1) and (2) ensure convergence of  $y$  towards  $g$ , and can be used to realize discrete point-to-point movements. To encode an arbitrary trajectory we modify (1) with a linear combination of kernel functions. A linear combination of kernel functions is given by

$$f(x) = \frac{\sum_{i=1}^N \omega_i \Psi_i(x)}{\sum_{i=1}^N \Psi_i(x)} x, \quad (3)$$

where

$$\Psi_i = \exp(-h_i (x - c_i)^2). \quad (4)$$

$c_i$  are the centers of the kernel functions, distributed along the trajectory, and  $h_i > 0$ . The variable  $x$ , which we call "the phase variable" is used in (3) instead of time. This way we avoid direct dependency of  $f$  on time. The dynamics of  $x$  is governed by

$$\tau \dot{x} = -\alpha_x x, \quad (5)$$

with an initial value  $x(0) = 1$ .  $x$  tends towards 0 as time increases. Here it is important, that  $x$  is not predefined but evolves over time with the integration of (5), and can be modified to account for perturbations and disturbances [14]. Considering (1) – (2) and (5) we now have a system of differential equations

$$\tau \dot{z} = \alpha_z (\beta_z (g - y) - z) + f(x), \quad (6)$$

$$\tau \dot{y} = z, \quad (7)$$

which can be used to approximate discrete movements. As  $x$  tends towards 0, the influence of the term  $f(x)$  decreases over time and system (6) – (7) converges to  $[0, g]^T$ , just as in (1) – (2).  $x$  also localizes the kernel functions along the trajectory that needs to be approximated. The resulting control policy, given with (5) – (7), associated with variable  $y$ , defines what is called a *dynamic movement primitive* (DMP).

## 3 GENERALIZATION

### 3.1 REPRODUCTION FROM A SINGLE DEMONSTRATION

Any smooth movement trajectory can be estimated by adapting the parameters  $w_i$  of (3). Triplets of sampled desired positions, velocities and accelerations are required,  $\{y_d(t_j), \dot{y}_d(t_j), \ddot{y}_d(t_j)\}, j = 1, \dots, T$ , where  $t_j$  are the sampling times. The trajectories can be obtained by kinesthetic guiding or human demonstration. The data can be in joint or Cartesian space and each degree of freedom is described by its own dynamic system. The two differential equations from (6) – (7) can be rewritten in a single equation by replacing  $z$  with  $\tau \dot{y}$  to get

$$\tau^2 \ddot{y} + \alpha_z \tau \dot{y} - \alpha_z \beta_z (g - y) = f(x), \quad (8)$$

where  $f$  is defined as in (3) and (4). We thus get

$$F(t_j) = \tau^2 \ddot{y}_d(t_j) + \alpha_z \tau \dot{y}_d(t_j) - \alpha_z \beta_z (g - y_d(t_j)), \quad (9)$$

$$\mathbf{f} = \begin{bmatrix} F(t_1) \\ \dots \\ F(t_T) \end{bmatrix}, \quad \mathbf{w} = \begin{bmatrix} w_1 \\ \dots \\ w_N \end{bmatrix}, \quad (10)$$

and obtain the following set of linear equations

$$\mathbf{Xw} = \mathbf{f}, \quad (11)$$

which needs to be solved to estimate the DMP describing the desired motion. For discrete movements, as is our case, the matrix  $\mathbf{X}$  is

$$\mathbf{X} = \begin{bmatrix} \frac{\Psi_1(x_1)}{\sum_{i=1}^N \Psi_1(x_1)} x_1 & \dots & \frac{\Psi_N(x_1)}{\sum_{i=1}^N \Psi_N(x_1)} x_1 \\ \dots & \dots & \dots \\ \frac{\Psi_1(x_T)}{\sum_{i=1}^N \Psi_1(x_T)} x_T & \dots & \frac{\Psi_N(x_T)}{\sum_{i=1}^N \Psi_N(x_T)} x_T \end{bmatrix} \quad (12)$$

The parameters  $\mathbf{w}$  can be calculated with the above system of differential equations in a least-squares sense. Such batch approach is common for discrete movements.

### 3.2 GENERALIZATION OF DISCRETE MOVEMENTS

All sampled trajectory points included in the training data set can be used. The number of training points is often large, so it is important to apply an efficient method to estimate them. Locally weighted regression (LWR) is a nonparametric regression method that fits local models to nearby data [1]. It has lower computational complexity than many other nonparametric regression methods, such as Gaussian process regression [11]. Using LWR and for a given query point  $q$ , the optimal parameters can be calculated directly from the available data by weighting the objective function

$$C(\mathbf{q}) = \sum_k L(\Xi(\mathbf{q}_k, \mathbf{w}), \mathbf{f}_k) K(d(\mathbf{q}, \mathbf{q}_k)). \quad (13)$$

Based on (11), local models are characterized by

$$L(\Xi(\mathbf{q}_k, \mathbf{w}), \mathbf{f}_k) = \|\mathbf{X}_k \mathbf{w} - \mathbf{f}_k\|^2. \quad (14)$$

Thus we need to minimize the objective function

$$\sum_{k=1}^M \|\mathbf{X}_k \mathbf{w} - \mathbf{f}_k\|^2 K(d(\mathbf{q}, \mathbf{q}_k)) \quad (15)$$

with respect to  $\mathbf{w}$ .  $K$  is in this case the kernel function, and  $d$  is the metrics in the space of query points  $\mathbf{q}$ . The weighting kernel is defined by

$$K(d) = \begin{cases} (1-|d|^3)^3 & \text{if } |d| < 1 \\ 0 & \text{else} \end{cases} \quad (16)$$

because this way it has finite extent and continuous first and second derivatives. Other types of kernels exist [1].  $K$  and distance  $d$  in the space of query points determine the influence of each example movement on the final estimate

of the control policy. The influence of separate example movements diminishes with the distance to the query point. If query points are in Euclidian space a standard Euclidian distance can be used

$$d(\mathbf{q}, \mathbf{q}_k) = \|\mathbf{D}(\mathbf{q} - \mathbf{q}_k)\|, \quad \mathbf{D} = \text{diag}(a_i), \quad a_i > 0. \quad (17)$$

We determined  $a_i$  so that at least two examples in each direction of the query point were relevant for the calculation of the new DMP.

The proposed approach is appropriate only if example trajectories smoothly transition as a function of query points. Otherwise nearby data does not provide information about the movement associated with the new query point  $q$ . Besides  $\mathbf{w}$  we also estimate the parameters  $\tau$  and  $g$ . This means that the function

$$\mathbf{G} : \mathbf{q} \rightarrow [\mathbf{w}^T, \tau, g^T] \quad (18)$$

which is in general unknown, needs to be smooth. This is usually the case if the robot uses the same strategy to solve the task in different situations.

### 3.3 OPTIMIZATION OF KERNEL FUNCTION LOCATIONS AND WIDTH

The accuracy of trajectory reproduction depends on the parameters of the DMPs. This is valid for simple reproduction of a single trajectory, and for the generation of new trajectories with the use of generalization.

One of the possibilities of increasing the accuracy of reproduction is with increasing the number  $N$  of kernel functions. This has unwanted side-effects, as it also increases the computational cost [6]. An even greater issue when increasing  $N$  is the width of the kernel functions  $h$ . A general rule that can apply is that for accurate reproduction there should be many and narrow kernel functions, and for a less accurate reproduction, they can be few and wide [14]. Another issue is the location of kernel functions. For a given  $N$  we can apply a linear distribution of the kernel functions over time, which results in the following distribution over  $x$ :

$$c_i = \exp(-\alpha_x \frac{i-1}{N-1}). \quad (19)$$

The width  $h_i$  of separate functions is determined by

$$h_i = \frac{2}{(c_{i+1} - c_i)^2}, h_N = h_N - 1, i = 1 \dots N. \quad (20)$$

In this paper we show how we can optimize the width and location of kernel functions for a predetermined number of kernel functions.

To choose the number of kernel functions we started with a small number and kept increasing it until the error of reproduction for a demonstration trajectory - at a set kernel function width and with a linear distribution of their locations - is below a selected threshold. This does not ensure the optimal number of kernel functions and higher accuracy could still be achieved. For example, it might happen that using 25 kernel functions would ensure better accuracy than using 20 kernel functions. But using 25 kernel functions with optimized width and locations would ensure an even greater accuracy, etc. So at one point we have to simply decide on a number.

To optimize the width and locations of kernel functions we minimize the criterion

$$\sum_{i=1}^M f_1(x_i), \quad (21)$$

where function  $f_1$  is determined with the average error of reproduction in discrete time samples

$$f_1(x_i) = \sum_{t=0}^T \overline{(y_i(t) - y_{i,demo}(t))^2} \quad (22)$$

$y_{i,demo}$  is determined with (5) – (7),  $i=1, \dots, M$  and  $M$  is the number of demonstration trajectories of the movement library.

Kernel function locations and widths have to be optimized for each of the degree of movement separately, i.e. if we generalize movement in joint-space of the robot, we have to perform optimization (and generalization) for each of the joints separately.

To minimize criterion (21) we use unconstrained minimization, which is for our functions computationally expensive. This is not a problem as we only have to run this procedure once (for each degree of movement) and can then use the results in the generalization of movements.

## 4 THROWING EXAMPLE

### 4.1 EXPERIMENTAL SETUP

To demonstrate the performance of the approach for the synthesis of a very dynamic task, we considered the task of throwing a ball to a target location with a robot. Robotic throwing has been studied in the past. Simple throwing was demonstrated already in [7], where the authors always performed throwing to the same target and not to a changing target as is the case in this paper.

Throwing depends on both the positional part of the movement and especially on the velocity and the release time of the ball. We considered planar throwing with the target location changing in two dimensions, namely the distance of the target from the robot and the height of the target. Because we can freely rotate the base of the robot, we can describe throwing in the entire 3D space. The experimental setup is shown in Figure 1.

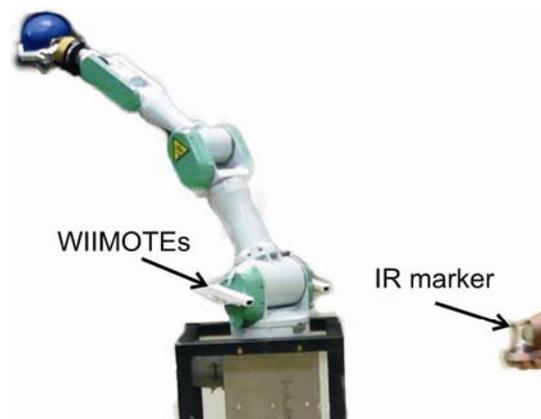


Figure 1 Experimental setup for the throwing example. 2 Wiimotes were attached to the Mitsubishi Pa-10 robot to act as cameras, which can detect an IR marker.

In our throwing experiment we used a 7 degrees of freedom Mitsubishi Pa-10 robotic arm. We used one degree of freedom for orientation of the robot and three joints for the throwing. To determine the target location we implemented a stereo 3D vision system that can detect and track an IR marker. For the vision system we used two calibrated Wiimotes, as is described in detail in [12]. The vision system was also used to determine the plane of throwing, i.e. to orient the robot towards the marker.

The target location, i.e. where the ball is supposed to land, is the query point into the library of demonstration movements. We recorded a set of 24 throwing examples within the field of view of the Wiimotes and within the range of throwing of the robot. We did not estimate the release time because the ball was not held firmly by the gripper, but detached from the hand due to the dynamics. The repeatability of the throwing was itself 2-3 centimeters. The radius of the ball was  $\sim 7$  centimeters.

### 4.2 EVALUATION

We evaluated the generalization of throwing by measuring the throwing error for a set of targets within the demonstration area. The results are presented in Figure 2, where the targets for the demonstration throwing are presented with stars and the targets for the evaluation with dots. The error of throwing was between -10 and +10 cm, while most of the area is within  $\pm 4$  cm, which is practically within the measuring and the repeatability error. Negative values represent too short throws and positive values too long throws.

The error of throwing is highest in the upper left corner of the evaluation area. This is because at that part the generalization algorithm takes into consideration mostly the demonstration trajectories for shorter throws. The number of trajectories taken into consideration for generalization is determined with the parameter  $a_i$  from (10), which we determined empirically. A different value could be used for the right-hand side of the demonstration area as there the demonstration trajectories are spaced further apart.

In fact, the demonstration trajectories were spaced evenly apart, but the vision system, which was calibrated up to two meters from the robot, returned incorrect values for targets further away. This proves not to be a problem as the demonstration points are associated with values from the same vision system as the query points. Even though the target is in reality closer to the robot, than what the vision system it is telling it, the algorithm compares to the values used in the demonstration, and throws at that distance. This effectively compensates for the error of the vision system.

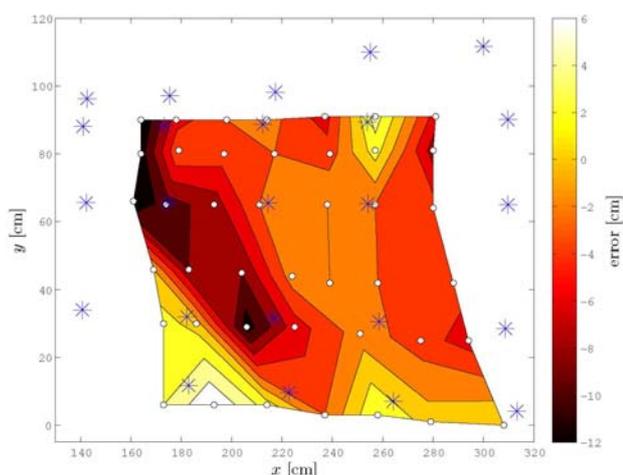


Figure 2 Accuracy of throwing. The stars are the training targets while the dots show the query points for generalization. For evaluation purposes the robot executed three throws at each target and we calculated the average throwing error from the desired target (query point). Negative values represent too short throws while positive values represent too long throws.

#### 4.3 OPTIMIZED VS. LINEARLY DISTRIBUTED KERNEL FUNCTIONS

In this section we show the difference between the use of optimized and linearly distributed kernel functions. The number of kernel functions was deliberately kept low at  $N=15$  so that the difference between trajectory approximations for the optimized and the linearly distributed kernel functions is more obvious.

It is important to keep in mind that the optimization tries to find the best solution for the entire set of demonstrated movements. This means that in average the results will improve, but it is not necessary that all the trajectories from the demonstration set will be approximated better.

As we can see from Figure 3, the optimization procedure changes the locations of the kernel functions. The kernel functions are spaced closer together at the section of the discrete movement, where most of the “action” is concentrated. For our case this section is the middle of the movement.

After the optimization the kernel functions are changed not only in location but also in width as can be observed from Figure 4. We can observe that the kernel functions are closer together and narrower in the middle of the movement

and further apart and wider at the start and end of the movement. This is in accordance with expectations, as most of the “moving” in the set of the demonstrated movements happens in the middle of the move. We should take into consideration that this kind of optimization and generalization are only applicable if all the trajectories have roughly the same characteristics.

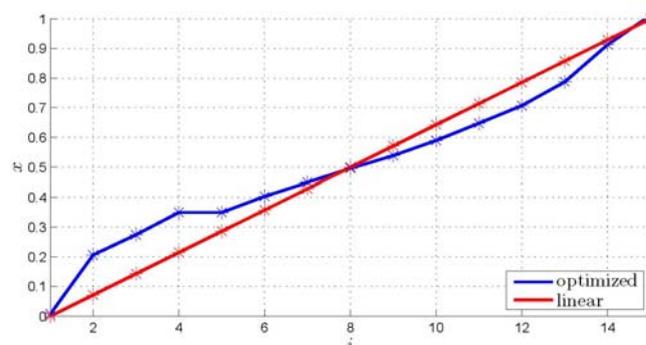


Figure 3 Locations of kernel functions for a linear distribution (red) and after optimization (blue).

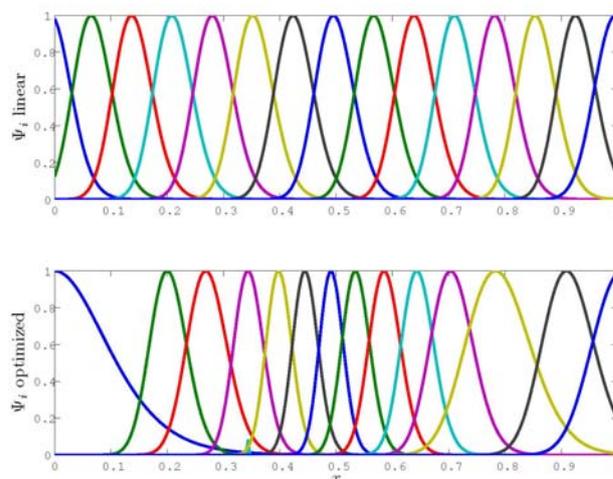


Figure 4 Kernel functions for linear (top) and optimized distribution over the course of the movement. The difference in width and locations is obvious. In the optimized distribution the kernel functions are closer together and narrower in the middle part of the movement, as there is where most of the “action” is happening.

Figures 5 and 6 show the difference in approximation of two trajectories from the demonstration set. Both trajectories show movement of the first robotic joint used for throwing in the experiment. Figure 5 shows the results for a throw relatively close to the robot and Figure 6 for a medium distance. Each figure shows the entire movement and two sections where the difference between the optimized and linearly distributed approximations is most obvious and critical. As we can see from both figures, the optimized trajectories ensure better approximation. The results are conditioned with a low number of kernel functions, and increasing the number of kernel functions would decrease the error of approximation.

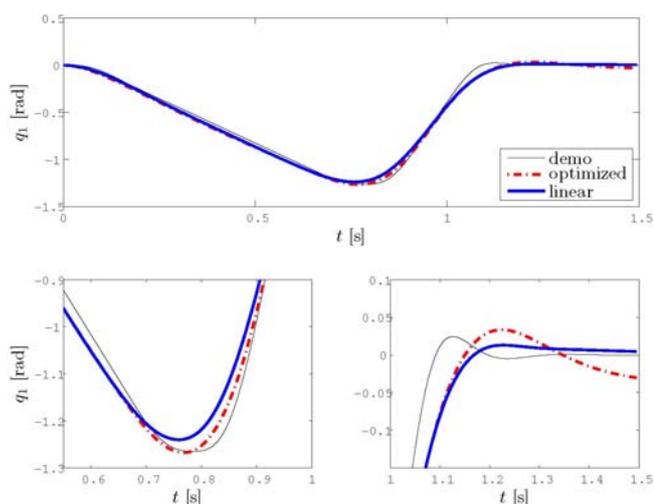


Figure 5 Difference between the demonstration example and approximations with linear and optimized kernel positions for a target close to the robot. The trajectory approximated with optimized kernel function locations and widths is presented with the dotted red line. The demonstration trajectory is presented with the thin black line. The approximation with a linear distribution of kernel functions is presented with a solid blue line. The trajectories are for only one joint of the robot.

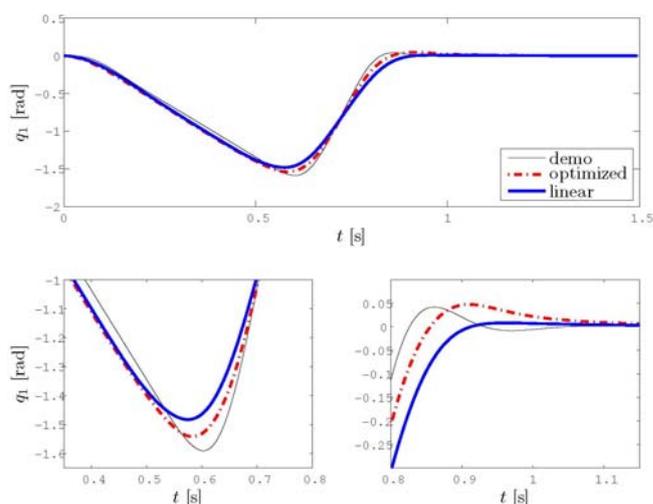


Figure 6 Difference between the demonstration example and approximations with linear and optimized kernel positions for a target at a medium distance from the robot. The trajectory approximated with optimized kernel function locations and widths is presented with the dotted red line. The demonstration trajectory is presented with the thin black line. The approximation with a linear distribution of kernel functions is presented with a solid blue line. The trajectories are for only one joint of the robot.

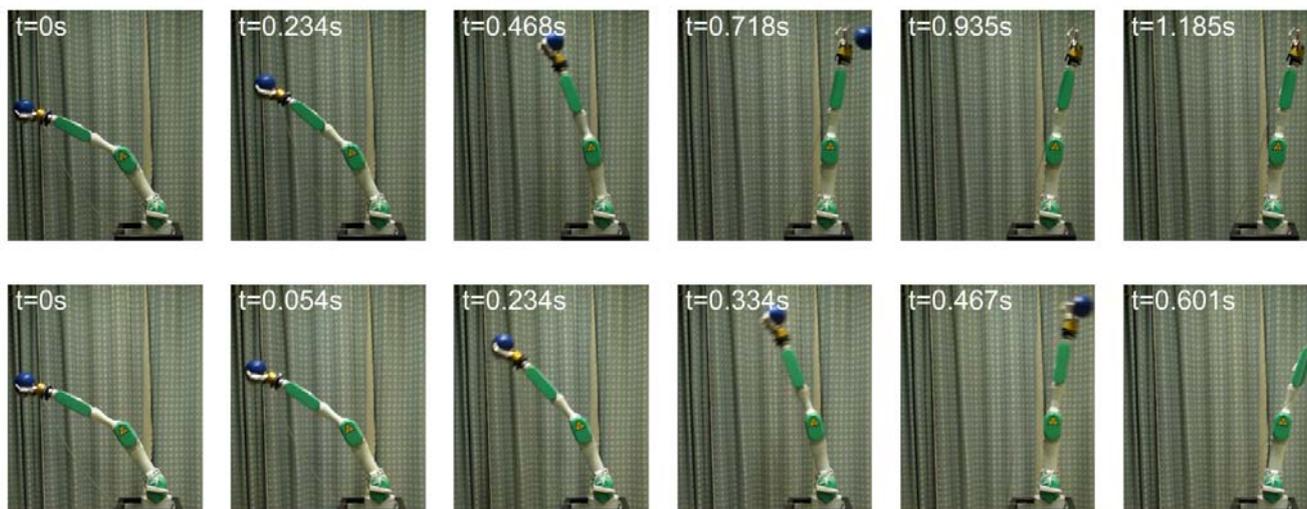


Figure 7 Sequences of still images of throwing at a target close to the robot (top) and far from the robot (bottom). Note the different time stamps which show that throwing further results in a much faster execution of the movement.

Figure 7 shows the execution of two throws for targets near to the robot (top) and further towards the edge of the throwing range (bottom). As we can see from the different time stamps, throwing further is executed much faster, and the robot reaches further. Note that the ball releases the hand because of the dynamics of the motion and not because it is released.

## 5 SUMMARY

In the paper we have shown effective use of a generalization algorithm to generalize a dynamic task of throwing, which has a very non-explicit connection between the query point and the final position of the movement. We performed throwing at desired locations with a robot. The accuracy of the throwing was in parts of the demonstrated area even better than our ability to

measure the landing spot. The accuracy can be further improved with changing the distance from the target for generalization ( $a_i$ ). In our case we could try different distances for short and long throws.

We expanded the generalization algorithm with an optimization of the kernel functions locations and widths, to achieve better approximation results. The results have shown a clear improvement of the approximation over the entire range of throwing when the locations and the width of the kernel functions were optimized.

Determining the appropriate number of kernel functions remains a problem and in the future we will have to rethink the solution also in light of the Locally Weighted Projection Algorithm [16].

#### ACKNOWLEDGEMENT

This paper was partially funded by the Slovenian Research Agency grant Z2-3672.

#### REFERENCES

- [1] Atkeson C.G., Moore A.W. and Schaal S., Locally weighted learning. *AI Review*, Vol. 11, pp. 11-73, 1997.
- [2] Calinon S., D'halluin F., Caldwell D.G. and Billard A., Handling of multiple constraints and motion alternatives in a robot programming by demonstration framework. *Proc. IEEE-RAS Intl Conf. on Humanoid Robots (Humanoids)*, December 2009.
- [3] Calinon S., Guenter F. and Billard A., On learning, representing and generalizing a task in a humanoid robot. *IEEE Transactions on Systems, Man and Cybernetics, Part B. Special issue on robot learning by observation, demonstration and imitation*, Vol. 37, No. 2, pp. 286-298, 2007.
- [4] Degallier S., Santos C.P., Righetti L. and Ijspeert A.J., Movement generation using dynamical systems: a humanoid robot performing a drumming task. *IEEE-RAS International Conference on Humanoid Robots (Humanoids06)*, 2006.
- [5] Gams A. and Ude A., Generalization of example movements with dynamical systems. *IEEE-RAS Intl. Conference on Humanoid Robots (Humanoids)*, 2009.
- [6] Gams A., Ijspeert A.J., Schaal S. and Lenarčič J., On-line learning and modulation of periodic movements with nonlinear dynamical systems. *Autonomous Robots*, Vol. 27, No. 1, pp. 3-23, 2009.
- [7] Hove B.M. and Slotine J.J., Experiments in robotic catching. *Proceedings of the 1991 American Control Conference*, 1991.
- [8] Ijspeert A.J., Nakanishi J., and Schaal S., Learning rhythmic movements by demonstration using nonlinear oscillators. *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Lausanne, Switzerland, 2002.
- [9] Ijspeert A.J., Nakanishi J. and Schaal S., Movement imitation with nonlinear dynamical systems in humanoid robots. *Proc. IEEE Int. Conf. Robotics and Automation*, Washington, DC, 2002.
- [10] Kawamura S. and Fukao N., Interpolation for input torque patterns obtained through learning control. *Proceedings of The Third International Conference on Automation, Robotics and Computer Vision (ICARCV'94)*, 1994.
- [11] Nguyen-Tuong D., Seeger M. and Peters J., Model learning with local Gaussian process regression. *Advanced Robotics*, Vol. 23, No.15, pp. 2015-2034, 2009.
- [12] Petrič T., Gams A., Ude A. and Žlajpah L., Real-time 3D marker tracking with a Wiimote stereo vision system: application to robotic throwing. *Proceedings RAAD 2010*, 2010.
- [13] Schaal S., Is imitation learning the route to humanoid robots? *Trends in cognitive sciences*, No. 3, pp. 233-242, 1999.
- [14] Schaal S., Mohajerian P. and Ijspeert A.J., Dynamics systems vs. optimal control - a unifying view. *Progress in Brain Research*, Vol. 165, No. 6, pp. 425-445, 2007.
- [15] Ude A., Gams A., Asfour T., and Morimoto J., Task-specific generalization of discrete and periodic dynamic movement primitives. *IEEE Transactions on Robotics*, Vol. 26, No. 5, pp. 800-815, 2010.
- [16] Vijayakumar S., D'Souza A., and Schaal S., Incremental online learning in high dimensions. *Neural Computation*, Vol. 17, No. 12, pp. 2602-2634, 2005.



# PNEUMATRONIC UNIT FOR MOTION OF BOWS

Enrico Ravina

Dept. of Mechanics and machine design, University of Genoa, Italy

## ABSTRACT

The paper describes the result of a theoretical and experimental research activity oriented to the design and realization of a low-cost pneumatic unit able to drive bows in relative motion respect to strings in bowed musical instrument. Pneumatic technology reduces costs and the proportional control allows the flexible programming of motion laws. The unit is specifically conceived to support laboratory tests of interest of violin makers and researchers. Results of specific experiments are collected and discussed.

Keywords: Mechanics, Pneumatics, Pneumatronics, Bows, Motion control

## 1 INTRODUCTION

The aim of design and construction of musical instruments is, of course, the achievement of desired sound results in terms of amplitude, tone, timbre and, more in general, of sound quality. Like any final phase of products construction also for musical instruments testing is foreseen. More in general, experiments and tests on stringed instruments need musicians to play: the result is consequently related to the characteristics of the player, because repeated actions are necessarily influenced by the human response. For this reason various devices and equipments conceived to play in automatic way (bowing machines) has been studied and proposed.

In particular, bowed instruments are played putting in relative motion the bow on the strings: in order to explore the performance of the built stringed instrument several different bowing gestures must be mechanically reproduced. In fact, different bowing gestures or *articulations* give the violin a range of different sounds. The differences are chiefly in the transient sounds at the beginning and end of the notes, and in the envelope: the way the sound varies over time.

In the middle of a long, sustained note, each vibration of the violin string and each cycle of the sound it produces are nearly identical to the one that preceded it. The string undergoes Helmholtz motion (steady state condition).

However, much of the interest in violin sounds comes from the transients: the short lived effects at the beginning and end of each note.

To the violinist, these are achieved by different articulations or bowing styles.

In particular:

- *vibrato*: consists on a small cyclic movement of one of the left fingers on the string - changes the pitch of the note played. It also changes the timbre. This timbre vibrato is very important to the characteristic sound of the violin;
- *tremolo*: in this movement the bow remains in contact with the string, but the bowing direction is changed rapidly. The rate of bowing is usually left to the performer so, in a violin or cello section, a tremolo note will have usually have bows moving in random phase;
- *sul tasto*: in Italian “sul tasto” means “on the fingerboard”: the string is bowed over the fingerboard, near the end. This position produces a sound with weaker high harmonics than normal playing (bow between fingerboard and bridge) and much weaker than other positions (e.g. *sul ponticello*). This condition is similar to *flautando*, which has a sound somewhat like that of a flute: less strong in high harmonics and with a little broad band sound as well;
- *sul ponticello*: in Italian “sul ponticello” means “on the bridge”. Bowing the string over the bridge, it is virtually impossible to set up stable, regular Helmholtz motion, and rather easy to excite, at least briefly, some harmonic Helmholtz motion. This gives a peculiar and irregular sound, showing a lot of high harmonics;

---

Contact author: Enrico Ravina

Dept. of mechanics and machine design (DIMEC)  
Faculty of Engineering  
University of Genoa  
Via Opera Pia 15 A, 16145 Genoa, Italy

- *spiccato*: the Italian word “spiccato” means “marked”. The bow lands percussively on the string and remains in contact while it is drawn across a little. The attack (the initial transient) is not as rapid as in *col legno*, because the bow hair is softer than the wood. Further, there is time for the bow hair to provide some continuous input of energy before it 'bounces' off the string;
- *pizzicato*: means “plucked with the finger”. The fleshy ball of the finger is used, rather than the nail. Once the string is released from the finger, there is no effective mechanism for putting more energy into the string (although violinists may try to prolong the sound by adding vibrato with the left hand). The corresponding sound has a moderately large magnitude initially, but decays rapidly away;
- *collé*: in French “collé” means 'glued'. The lower part of the bow (which can exert more force) strikes the string rapidly. The sound builds up rapidly at the start of each note, and then slows smoothly before lifting off;

- *col legno*: is an unusual articulation. In Italian “col legno” means “with the wood” and that is just what the violinist does, playing the string with the wood of the back of the bow and making a rattling sound on the string. The magnitude of the sound then decreases rapidly, because the wood does not input energy effectively as it is dragged over the strings.

Examples of acoustic responses corresponding to different articulations are collected in Table I.

The testing must be oriented to check the performance repeatability under cyclic relative motions between bow and string. Problems related to mechanical bowing of string instruments have been studied by several researchers, in order to eliminate the subjective human influence on the sound produced. Different mechanical devices have been proposed in order to substitute players with automatic units. Raman (1920) moved the violin beneath the bow, Saunders (1937) suggested mechanisms with actuated celluloid discs, Meinel (1937) and Rohloff (1940) proposed McLennann (1995) compact and continuously moving belts electrically driven. More recently mechanized and electronically controlled devices has been proposed and cited at paragraph III.

The solution discussed hereafter is based on a pneumatic actuation system, electronically controlled, able to drive an actual bow: its mechatronic features allow an easy and flexible programming of the motion law.

Table I - Time histories corresponding to different articulations

Articulation	Acoustic Response
Vibrato	
Tremolo	
Sul tasto	
Sul ponticello	
Spiccato	
Pizzicato	
Collé	
Col legno	

## 2 PROBLEM FORMULATION

Relative motion between bow and strings is related to the attacks of the bow: specific studies about measurement of bow motion and bow force in violin playing (Askenfelt, 1986). In addition, instantaneous velocities and accelerations are variables unconsciously by players and could be managed by bowing machines.

Bow motion is not only related to the musical technique but to mechanical and physical phenomena related to the contact between horsehairs and violin strings: these phenomena depend on the size and shape of bows and violins, and to the local forces applied by the player by means its wrist and indirectly from bow to string. A flexible approach able to manage gradients of velocities and accelerations and to instantaneously modify forces is reached applying pneumatic technology.

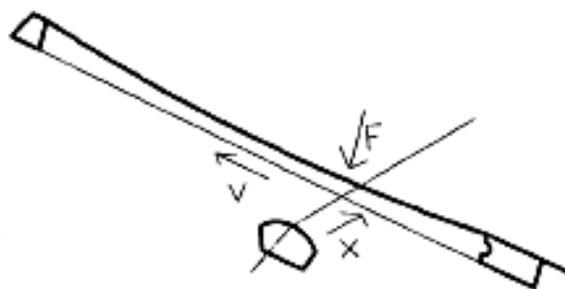


Figure 1 Fundamental variables.

The main problem is the governor of the bow, including structural, kinetic and dynamic aspects. Fig. 1 sketches the fundamental variables: force applied on the string ( $F$ ), distance between bridge and contact ( $x$ ) and bow velocity ( $v$ ). Distance and velocities depend on the articulation of the bow on the strings. Force is imposed by the wrist of the musician. In order to define the low motion it is necessary to know the 3D trajectories of the bow: Hodgson plots are used in the present study [1]: these plots are usually available in orthographic back projection ( $x$ - $z$  plane and  $x$ - $y$  plane). Fig. 2 reports an example of these plots.

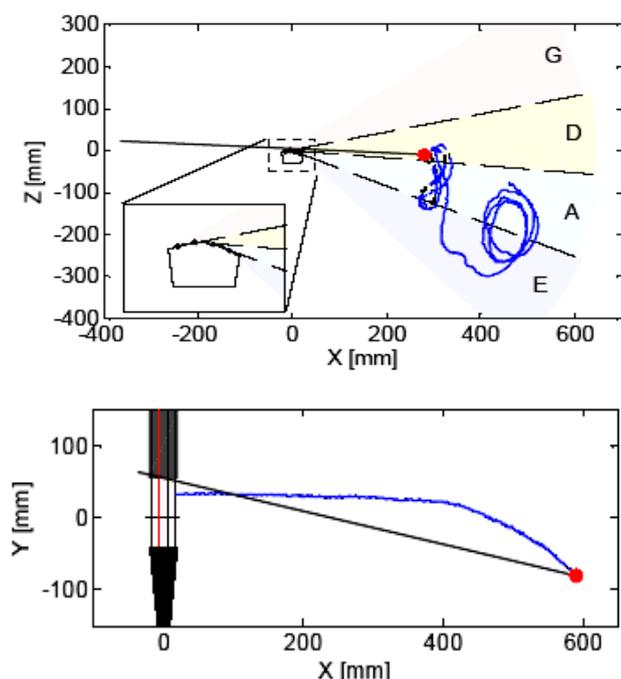


Figure 2 Examples of Hodgson plots.

The red dot indicates the position of the frog at the “present” moment (i.e., at the end of the selected time interval). The black line corresponds approximately to the bow-hair ribbon from the frog to the tip, ignoring the bending of the hair at the bow-string contact point. The trajectory history of the bow frog is indicated by a blue line, shown as solid and fat when the bow was in contact with the string, thin and dotted otherwise. In the background, the bridge, string positions and string crossing angles are shown (see close-up for more detail), forming the functional context of the displayed bowing gestures. The string crossing angles (dashed lines) subdivide the space into 4 angular zones associated with the bowing of the different strings. The zones are indicated with different colours: blue (E string), green (A string), yellow (D string) and red (G string).

This approach has been followed in various specific researches involving gestures in musical events [2, 3, 4, 5 and 6].

From the mechanical point of view, the string displacement can be modelled through the equation:

$$y(x,t) = \sum_{n=1}^{\infty} a_n \sin \frac{n\pi x}{L} \sin n\omega t \quad (1)$$

Maximum string displacement  $y_m$  and maximum force on the bridge are related to the ratio  $v_b/x_b$ :

$$y_m = \frac{1}{8f} \frac{v_b}{x_b} \quad (2)$$

$$F_m = \mu c \frac{v_b}{x_b} \quad (3)$$

where  $v_b$  is the bow velocity,  $x_b$  is the distance between the contact point of bow and string from the bridge,  $f$  is the frequency,  $\mu$  the mass for unit length,  $c$  the wave velocity propagation and the product  $\mu c$  represents the characteristic string impedance.

This variable must be carefully considered designing the motion control of an automatic unit. Mechanics of contact between bow and strings has been subject of specific studies [7].

### 3 STATE OF THE ART

Equipments playing violins or bowed instruments are proposed by different researchers: Sobh et al. [8] developed experimental robot musicians, where a violin is set up with two bows controlled by two servos and twelve fingers (solenoids). Servos are attached to bows of the violin. They move the bows back and forth across the violin strings.

Chia et al. [9] proposed “RoboFiddler”: the main aim of that project was to design and build a robotic system that plays the violin correctly and accordingly to a given set of musical notes. The project design centres on the real violin, where the playing techniques and tone production were closely researched and adopted.

The physical features of the violin mainly determine the design dimensions of the prototype. It includes the bowing arm system for the violin right hand bowing technique, the fingering mechanism for the violin left hand fingering technique, and the control system for driving the actuators. The bowing arm system uses two servo motors to control the bowing angle and pressure on the strings respectively. It also uses a DC motor to control the bow speed and direction.

Therefore the system actuates in a three degree of sliding, tilting, and lifting motion, where the bow interacts with the violin string, causing it to oscillate thus producing a true violin sound.

Finally humanoid robots able to manipulate bows and play violins are studied, designed and realised in Japan [10].

In general three relative motions between bow and strings must be performed: bow translation, rotation (modification of the attack angle) and different distance from the bridge. A low cost solution is described hereafter.

#### 4 EXPERIMENTAL SETUP

The proposed unit is essentially based on pneumatic actuators, electronically controlled. Main advantages with respect to electric servo systems are essentially identified in:

- low-cost, simple and reliable technology;
- operation in very similar conditions to actual ones;
- low noisy;
- simplified programming.

The linear motion of the bow is generated by means a linear pneumatic actuator driven by flow-control proportional valve. Position feedback is obtained by magneto-strictive transducer embedded to the actuator.

The proportional valve is controlled by a dedicated electronic unit interfaced to PC. Fig. 3 reports an overall view of the unit. The instantaneous attack angle and the relative distance between bow and bridge are modified by a swivel-linear pneumatic actuator (Fig. 4): both motions can be actuated and controlled separately or simultaneously: in the present application first solution is recommended.



Figure 3 Pneumatronic unit.



Figure 4 Swivel-linear actuator.

Main features of this device are:

- high repetition accuracy thanks to cushioning components with fixed stop (1);
- swivel angle can be infinitely and accurately set;
- mechanical gearing between the stop element and swivel;
- prevention of movement of the stop system under load (2 by position sensing (2), sensor retainer (3) and stop lever (4));
- compact sensing of the swivel motion via proximity sensors (by piston rod (5), slot for proximity sensor (6) and precision end position (7));
- adjustment angle scale (8);
- swivel motion of up to 270°;
- linear motion of up to 200 mm stroke;
- high rotational energy during swivelling thanks to directly mounted, self-adjusting shock absorbers;
- supply ports at one end for quick, clear-cut tubing connections;
- high precision thanks to re-circulating ball bearing guide;
- compact sensing of the linear position using proximity sensors;
- precision adjustment of the end positions, after the lock nut is loosened;
- easily presetting of the required angle using the scale.

This device supports axially the instrument and allows both the rotation and the variation of distance between bridge and bow. Details on the connections to the musical instrument are shown in Fig. 5.



Figure 5 Connections between instrument and swivel-linear actuator.



Figure 6 Details of bows.

Another significant mechanical aspect concerns the generation of the force applied to the bow, strongly influencing the contact force between bow and strings. As previously cited, this force is really generated by the wrist of the player; in the proposed pneumatronic unit it is produced by a spring, embedded on the original frame

designed to support the frog (Fig.6), taking into account the evolution of forms and dimensions of this component from baroque to modern design.

The difference almost certainly results from the different ways in which the bow is held in these instrument families: violin/viola/cello players hold the wood part of the bow closer to the palm, whereas gamba players use the opposite orientation, with the horsehair closer. The orientation appropriate to each instrument family permits the stronger wrist muscles (flexors) to reinforce the strong beat.

Fig. 7 reports details of the proposed mechanical wrist and its connection to other components of the unit. The elastic force can be adjusted modifying the relative position of the two ends of the spring. In this way after each motion cycle the force can be changed.



Figure 7 Details of the bow support.

An improvement involving pneumatic muscles is under development: the goal is to control in real time also the generated force, correlating the actual force to the bow motion. A view of the assembled unit is shown in Fig. 8.

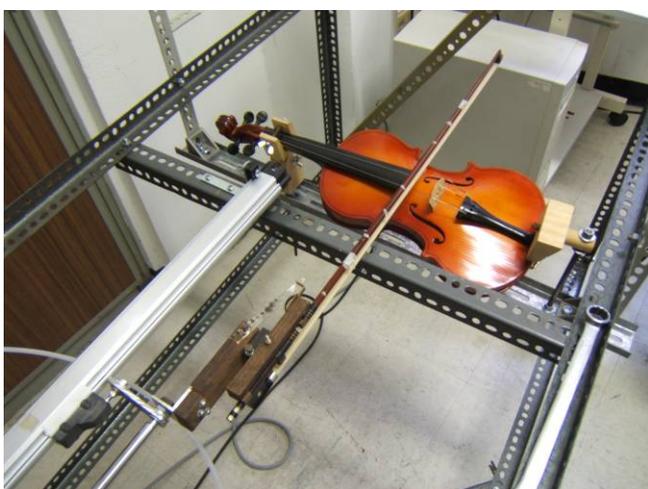


Figure 8 Overall view of the unit.

## 5 MOTION RESULTS

The motion of the bow is generated through DZH 32-500 pneumatic cylinder (by FESTO): the elliptic shape of the piston avoids uncontrolled axial rotations. It is driven by MPYE 5-1/8" flow-control proportional valve (by FESTO). Position feedback is realized by external magneto-strictive position transducer. Details of these devices are collected in Fig. 9.



Figure 9 Details of pneumatic devices.

Improvements of this architecture are, at present time, under development, involving rod less pneumatic axis with integrated position transducer.

The electronic control is committed to SPC100 controller (by FESTO), interfaced to PC (Fig. 10). This unit is able to compare theoretical and actual positions generating the electric signal to proportional valve. In addition, the dynamic response is optimized thanks to selection of internal control parameters and to internal velocity and acceleration feedbacks.



Figure 10 SPC100 controller.

Table II - Example of programmed sequence

Statement	Description
N0 G62 X	Quick Stop
N10 G08 X10	Acceleration ramp to 100% of the max. value
N20 G09 X10	Deceleration ramp to 100% of the max. value
N30 G01 X100 FX1	Move at 100 mm with 10% of the max. speed
N40 G01 X150 FX8	Move at 150 mm with 80% of the max. speed
N50 G01 X100 FX6	Move at 100 mm with 60% of the max. speed
N60 G01 X150 FX6	Move at 150 mm with 60% of the max. speed
N70 G01 X100 FX7	Move at 100 mm with 70% of the max. speed
N80 G01 X150 FX7	Move at 150 mm with 70% of the max. speed
N90 G01 X100 FX8	Move at 100 mm with 80% of the max. speed
N100 G01 X 150 FX8	Move at 150 mm with 80% of the max. speed
N110 G01 X100 FX7	Move at 100 mm with 70% of the max. speed
N120 G01 X150 FX7	Move at 150 mm with 70% of the max. speed
N130 G01 X100 FX6	Move at 100 mm with 60% of the max. speed
N140 G01 X150 FX6	Move at 150 mm with 60% of the max. speed
N150 M30	Repeat

The device is interfaced and synchronized to the digital control related to the axial motion of the instrument under test. The motion law can be programmed as function of the articulations to be simulated. Programming code is particularly user-oriented: an example of code is collected in Tab. II.

Results of cyclic motion laws are reported in Fig. 11: nominal values are compared to actual values and detailed analyses about motion errors or local instability phenomena (Fig. 12) can be performed.

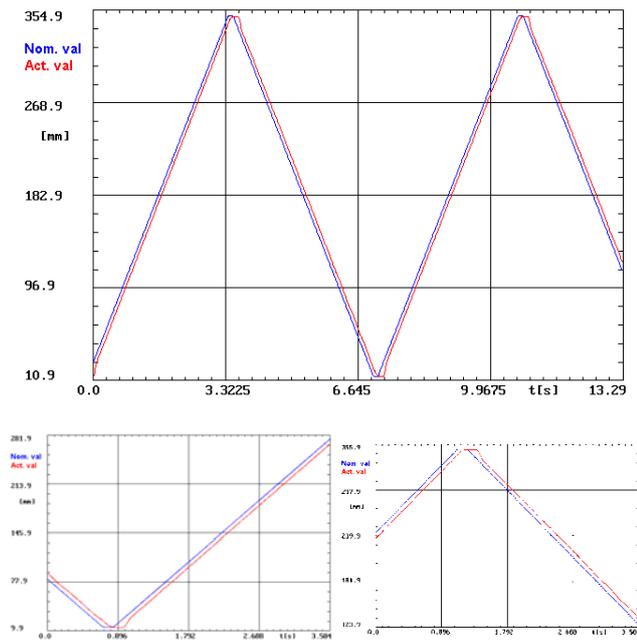


Figure 11 Example of dynamic responses.

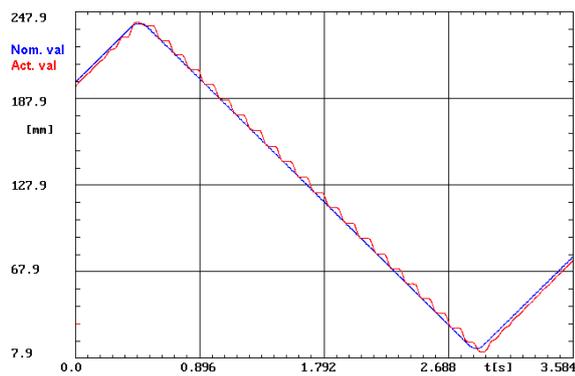


Figure 12 Instability phenomena.

## 6 ACOUSTIC RESULTS

In order to validate the performance of the proposed mechatronic equipment acoustic acquisitions of sound generated by violins played by a musician are compared to the corresponding acquisitions under mechanical actions. Fig. 13 shows the acoustic time history and the corresponding frequency response function acquired on a standard violin played by a musician on the D4 note.

Fig. 14 collects the same results playing the violin by the mechatronic equipment, emulating the sequence of motions generated by the player. Time histories are only apparently different: the comparison of FRFs show a good correspondence of results.

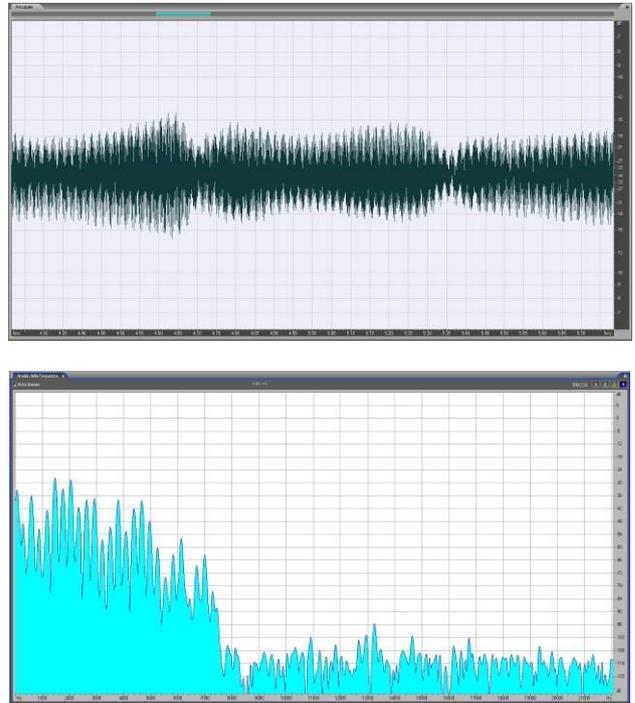


Figure 13 D4 played by musician.

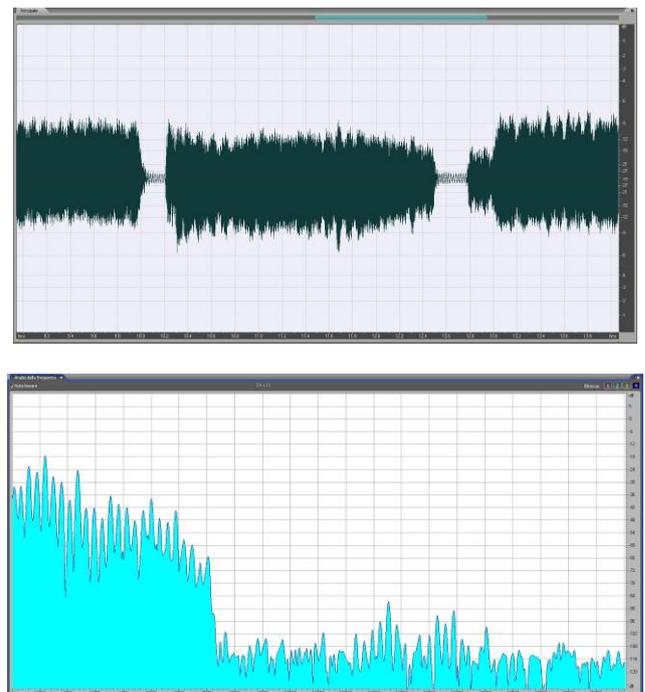


Figure 14 D4 played by pneumatronic unit.

## 7 CONCLUDING REMARKS

The proposed pneumatronic unit integrates the advantages of the pneumatic technology with ones of the electronic control to generate emulated relative motions between bow and strings in bowed musical instruments. Pneumatic proportional control has shown very good flexibility and the optimization of set of control parameters (in particular gain, and damping) allows achieving very sharp motion laws. The unit has been tested on various instruments of the family of violins showing a good response with respect to actions generated by musicians.

Further improvements, at present time under developments, will involve rod less actuator for primary motion of the bow and fluidic micro-muscles in the real time force control in the contact between bow and strings.

## REFERENCES

- [1] Schoonderwaldt E. and Wanderley M., Visualization of bowing gestures for feedback: the Hodgson plot. *Proceedings of the 3rd International Conference on Automated Production of Cross Media Content for Multi-channel Distribution (AXMEDIS)*, Firenze University Press, pp. 65-70, 2007.
- [2] Baiadera A.P., Kazennikov O. and Wiesendanger M., Coordination of bowing and fingering in violin playing. *Cognitive Brain Research*, Vol. 23, pp. 436-443, 2005.
- [3] Maestre E., *Bowed-string instrumental gesture analysis and modeling for sound generation*. 2nd EyesWeb Week, Genoa, February 2008.
- [4] Van der Linden J., Schoonderwaldt E. and Bird J., Good Vibrations: Guiding Body Movements with Vibrotactile Feedback. *Proceedings of the Third Workshop on Physicality*, Cambridge, U.K., 2009.
- [5] Ng, K., Larkin O., Koerselman T. and Ong B., 3D Motion Data Analysis and Visualisation for String Practice Training. *EVA London Conf.*, 11-13 July 2007.
- [6] Kojs J., The Language of Action and Cyberaction-based Music: Theory and Practice. *Journal of New Music Research*, Vol. 38, No. 3, pp. 285-294, 2001.
- [7] Pitteroff R. and Woodhouse J., Mechanics of the contact area between violin bow and a string. *Acta Acustica*, Vol. 84, pp. 929-946, 1988.
- [8] Sobh T.M., Wang B. and Coble K.W., Experimental robot musicians. *Journal of Intelligent and Robotic Systems*, Vol. 38, pp. 197-212, 2003.
- [9] Chia Y.C., Hong B.Y., Lee C.H. and Lim B.K., *RoboFiddler*, Level IV Project, Final Report, University of Adelaide, Australia, 2006.
- [10] Kajitani M., Development of musician robots in Japan. *Australian Conference on Robotics and Automation*, Brisbane, Australia, 1999.



# A MULTI-BEHAVIOUR ALGORITHM FOR AUTO-GUIDED MOVEMENTS IN SURGEON ASSISTANCE

E. Bauzano, V.F. Muñoz, I. Garcia-Morales

University of Malaga, Spain

## ABSTRACT

This paper focuses on autonomous movements to aid the surgeon to perform certain tasks. Robotic assistants have solved the drawbacks of Minimally Invasive Surgery (MIS) and provide additional skills to the surgeons. However, some authors argue that these systems could lengthen the operating time. The solution is the automation of certain maneuvers that help the surgeon during a surgical maneuver. This work proposes control architecture for a surgical robot capable of performing autonomous movements. In this way, a trajectory planner based on a behaviour concept computes the required velocity vector of the surgical instrument hold by the robot.

Keywords: Automatic Movements; Minimally Invasive Surgery; Robot Assistant

## 1 INTRODUCTION

The enormous complexity and costs limit the clinical impact of the robotic assistants for Minimally Invasive Surgery (MIS) despite of its well known advantages. Some authors argue that the use of robots in surgery, although providing more precision, could lengthen operating time [1].

One solution to this problem consists of automating certain surgical maneuvers. *Visual Servoing* is one of the most common techniques to perform automated tasks. Control of surgical instrument movements involves calculating instant linear and angular velocity references in each control period. These references are obtained by analyzing the images from a non-calibrated stereo vision system [2] or acquiring those images through a regular laparoscopic surgery camera [3]. Visual Servoing also enables safe movements of the laparoscopic tool, for example, on cardiac surgery, so the instruments are synced with the heart beat [4]-[6].

Other works are devoted to assist the surgeon with robots during the intervention. This way, some developments have performed autonomous stitching and knot tying procedures [7]; others automatically guide a robotic tip in colonoscopy

[8], provide automatic transformations to a robot assistant from laparoscopic navigation to open-surgery motion [9], or give autonomous decisions on *teleoperation* with high communication latency or low bandwidth [10]. There are also more complex systems like EndoPAR which automates a knot tying procedure in heart surgery [11], and allows the surgeon to operate as if there was no heartbeat. Furthermore, there are also studies for human-machine skill transfer on robot assistants so they may perform automatic knot tying procedures [12], or automatic navigation on cholecystectomy without pre-operative information [13].

This paper proposes a solution for an automatic movement of the surgical instrument arm equipped with an active wrist. A passive wrist emulation (PWE) controller is needed in order to know the point of insertion so-called *fulcrum points*, as well as minimize the forces over the abdomen. This arm must also avoid the collision with the surgeon tool during its navigation. The main goal of this development is devoted to replacing a human assistant for the surgeon in certain laparoscopic surgery procedures. In this way, authors propose a control architecture for performing automated tasks which allows adapting the movements to the surgeon actions.

The structure of this article is divided into six sections. After this introduction, section 2 states the control architecture proposed to solve auto-guided movements on MIS robotic assistants, whereas section 3 resumes the control philosophy of the PWE to locate the fulcrum point.

---

Contact author: Enrique Bauzano Nuñez

E-mail: ebauzano@isa.uma.es

With this general solution, section 4 explains the developed methodology for moving the robot to one of the surgeon's tool tip. This technique has been applied to the task of taking gauze to the surgeon, as presented on section 5. Finally, section 6 discusses some possible improvements to the methodology, as well as related future works.

## 2 CONTROL ARCHITECTURE

This section introduces the architecture proposed to perform automatic movements. Firstly, a brief description of the surgical workspace is given to explain the situation where the robot has to develop the task, as well as all movement constraints that limit its freedom. Once the problem has been presented, the specifications the robot must accomplish are discussed. These requirements should let the robotic assistant navigate into the abdominal cavity in a safely way.

### 2.1 PROBLEM STATEMENT

The environment where the robot interacts with both the patient and the surgeon consists of a closed space, the abdominal cavity, as it is shown in Figure 1. The camera  $C$  and the instrument of the robot  $R$  are inserted through their respective fulcrum points  $G_C$  and  $G_R$ . Moreover, this environment also includes the surgeon's tools used for surgery procedures. One of them is considered the target for the robot  $S$ , whereas the other one is the obstacle  $O$ .

The abdominal cavity where the robot instrument may move is defined by a cone-shaped view field, which contains the scene seen in the screen by the surgeon. The robot can just analyze this space, so both, its instrument  $R$  and the surgeon's tool  $S$  must remain inside of this cone. This situation may be fulfilled by following the surgeon's tool  $S$  with the camera and forcing the robot tool  $R$  to be within the view field. As an additional hypothesis, the surgeon is able to displace his or her tools  $O$  and  $S$  during any movement of the robot. This work supposes that the visual servoing problem is solved.

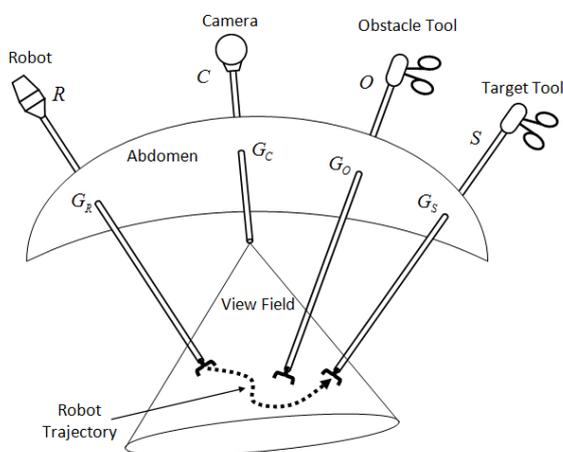


Figure 1 The camera is focused over the surgical workspace, while robot tool goes where the surgeon's target tool is located.

The fulcrum point  $G_R$  is a movement constraint for the robot additional instrument  $R$ . Therefore, movements inside the patient are limited to four degrees of freedom. In this way, in order to control the position of the surgical instrument, it is used a motion controller to take into account these holonomic constraints [14].

This paper suggests moving the robot instrument  $R$  to the surgeon's target tool  $S$  location. The surgeon tool  $O$  is defined as the unique obstacle to be avoided during an automatic task. The trajectory has to be calculated on-line, because the surgeon's tools are not static since he or she continues the intervention normally. All these considerations will be taken into account on the control architecture proposed on next subsection.

### 2.2 ARCHITECTURE SCHEME

Once the functionality needed for the robot has been described, an architecture scheme which resumes the capabilities of the system can be introduced as shown in Figure 2. The main element is the *Local Planner*, whose mission is to guide the robot instrument to the surgeon tool avoiding the obstacles.

For this purpose, some information is needed from the environment. In addition to the own kinematics of the robot given by the *Robot Location* feedback, the *Surgeon Location* must be known in order not only for the current tool position, but also for estimating its trajectory thanks to the velocity and acceleration parameters. This prediction is done by the *Surgeon Trajectory Estimator* through the obstacle tool velocity  $\vec{v}_O$ , and allows the robot to update its trajectory  $\vec{v}_R$  in order to avoid possible collisions with the surgeon's obstacle tool  $O$ .

With all this data, the *Local Planner* may command the special instrument through the *Passive Wrist Emulation Control*. This element is necessary to know the fulcrum point, and generates the Planned Trajectories for the *Robot Arm* spherical navigation. As the robot has an active wrist, the Passive Wrist Emulation (PWE) allows minimizing the forces over the abdomen of the patient [14]. This controller is explained on next section.

## 3 PASSIVE WRIST EMULATION ON ACTIVE WRIST

Direct actuated wrists avoid the backlash introduced by the trocar-instrument interaction which appears on passive wrists solutions. Thus, the endoscope always reaches the desired spherical position in spite of inaccurate estimation of the fulcrum point. However, this uncertainty will eventually force the abdominal wall and could damage the patient [14]. This section will resume how simultaneously estimate the fulcrum point while reducing the force applied over the abdomen.

Figure 3 illustrates an endoscope movement where the altitude angle  $\beta$  changes from a starting null position  $A$  to a final position  $B$ . The robot movement planner uses an initial estimation of the fulcrum position  $C$  for computing the arc shape trajectory, instead of the unknown real one  $I_0$ .

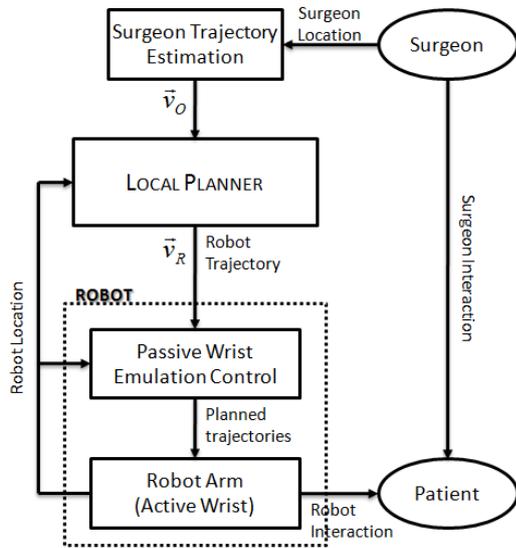


Figure 2 Architecture scheme for performing automatic tasks of a robot assistant.

In such way, the abdominal force  $\vec{F}_a$  gives information about the direction where the fulcrum point  $I$  is displaced. Therefore, this force is used for two purposes: i) if navigation is being performed in an accurate way; ii) any unexpected change implies a fulcrum location variation (i.e. respiratory motion or abdominal cavity air pressure variation).

The proposed approach to reduce the efforts applied to the abdominal wall is based on emulating the passive wrist behaviour. On the situation shown in Figure 3, a non-actuated wrist would keep its orientation according to the initial position at the fulcrum point  $I_0$  by means of the angle  $\varphi$ . Thus, the motion controller computes  $\varphi$  thanks to a spring model of the abdomen where both, the force  $\vec{F}_a$  and the abdomen stiffness are known [14].

#### 4 AUTO-GUIDED METHODOLOGY

The automated movement proposed in this paper consists of reaching a goal position defined by the surgeon's target tool tip  $S$ , as it was stated in Figure 1. In this way, the other surgeon's tool  $O$  is considered an obstacle the robot must avoid. As an additional hypothesis, both surgical tools may be displaced during the robot movement.

Figure 4 shows the proposed *Local Planner* scheme stated in Figure 2 which has been used to compute the required velocity  $\vec{v}_R$  for the robot tool ( $R$  in Figure 1) in order to reach the target tool  $S$  by avoiding the obstacle  $O$ . The current velocity of the robot  $\vec{v}_R$  and the obstacle tool  $\vec{v}_O$  are used on a Fuzzy Logic algorithm for deciding the best strategy to get closer to the target without collisions.

This work proposes the combination of three behaviours in order to plan the required robot velocity:

- The “*Go-to Target*” behaviour finds a trajectory to the target avoiding static obstacles using the Artificial Potential Field methodology (APF), but cannot deal with movable obstacles. However, it can react to changes on the target location.
- The “*Velocity Corrections*” behavior will change the robot velocity  $\vec{v}_R$  depending on the obstacle velocity  $\vec{v}_O$ . These corrections will be more important with high velocities and when the robot is nearby the obstacle tool.
- The “*Backward Movement*” behaviour covers the possibility that the robot tool and the obstacle tool are so close they may collide. This situation is solved by adapting the robot movement to the obstacle in order to avoid the collision, as the surgeon will probably want to displace the robot to free his or her vision space.

As it has been already stated, the decision of which behavior is the best on each situation is taken by a Fuzzy algorithm. Depending on the directions of the robot velocity  $\vec{v}_R$  and the obstacle one  $\vec{v}_O$ , the Fuzzy decision will use the best combination of the behaviours previously stated. The truth table of this Fuzzy decision appears on Table I.

Table I - Truth Table for Behavior Decision

$\vec{v}_O \backslash \vec{v}_R$	1	2	3	4
1	A	FA	HC	C
2	FA	A	C	HC
3	HC	C	A	FA
4	C	HC	FA	A

A = go Away, FA = Far Away, C = get Closer, HC = High Close.

In this table, velocities have labels assigned. Each label indicates the quadrant of a circle where the velocity  $\vec{v}_R$  and

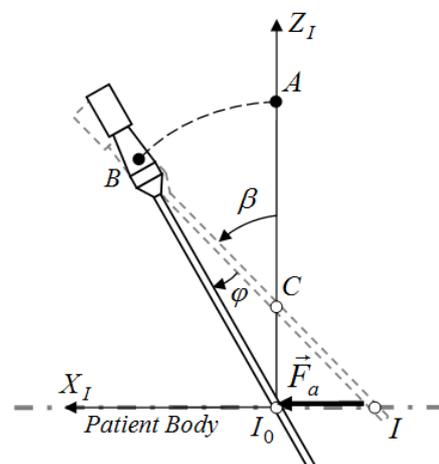


Figure 3 Passive Wrist Emulation behaviour.

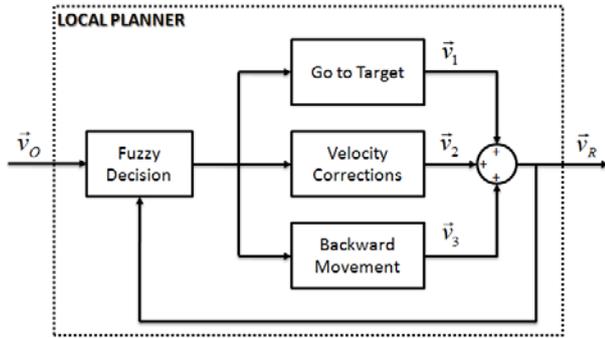


Figure 4 Local Planner scheme proposed to avoid dynamic obstacles.

$\vec{v}_O$  are orientated in terms of a reference axes. For example, if both velocities are 1 then their directions are the same, whereas if  $\vec{v}_R$  is 1 and  $\vec{v}_O$  is 3 mean that those directions are opposed. The output of Table I is a fuzzy value that is used to select the importance of each behaviour.

#### 4.1 THE “GO-TO TARGET” BEHAVIOUR

This behaviour is devoted to find a path to the target tool avoiding static obstacles. This work has applied the Artificial Potential Field (APF) methodology to fulfill this task. APF associates a repulsive potential field  $U_{rep}$  for each obstacle of the environment, as well as an attractive potential field to the target  $U_{att}$ . Thus, expression (1) states that the resulting potential field generates a virtual force  $\vec{F}_1$  which both, attracts the robot to the goal with force  $\vec{F}_{att}$  and repels it from the collision with the obstacles through  $\vec{F}_{rep}$  force.

$$\vec{F}_1 = -\nabla U_{att} - \nabla U_{rep} = \vec{F}_{att} + \vec{F}_{rep} \quad (1)$$

The virtual potential functions of this work are based on the expressions of [15]. They have a high value only on surrounding obstacle area, whereas the attractor has a potential field which is proportional to the distance from the robot to the target. The gradient of these expressions gives the force generated by the potential field.

The main problem of APF methodology appears by means of local minima points that may be found on the workspace. This work extends the potential field of [15] by applying the solution proposed by [16] and uses a force field capable of escape from these local minima points.

The APF methodology is commonly applied to path planning of point mobile robots. However, MIS problem requires that the robot  $R$  is not just a point, but a line which departs from the fulcrum point  $G_R$  to its tool tip, as it has already been shown in Figure 1. One consequence is that the target also becomes a line which departs from the robot fulcrum point  $G_R$  and ends on the surgeon target tool tip. Furthermore, as both instruments are long enough to be considered like one-dimensional objects, they may only

collide in one point. As shown in Figure 5, the minimal distance  $\rho$  between the robot tool and the obstacle tool defines the point over each tool  $M_R$  and  $M_O$  that would collide in case they approach. Point  $M_R$  is called the guide point, because it is used to move the robot to the target.

Secondly, the default potential function generates equipotential surfaces with a cylindrical shape. However, this work has chosen the use of conical surfaces with their vertices on the fulcrum point of the obstacle, as it can be shown in Figure 5. The reason is that the movements are faster near the tool tips, so higher potentials are needed to maintain the distance between robot and obstacle tool.

Therefore, the APF methodology can be just applied to the guide point  $M_R$  instead all the robot tool longitude. This point has a target associated to its final position  $M_R^f$  (see Figure 5) when the instrument has reached the target. Thus, to calculate the next trajectory location the algorithm steps are:

#### Procedure for Automatic Movements

- 1) Locate the minimal distance point  $M_R$
- 2) Calculate its target  $M_R^f$
- 3) Apply the forces given by (1) in order to know the needed velocity of the guide point
- 4) Move the robot instrument to fit the new location and the fulcrum point constraint

#### End Procedure

First step can be geometrically deduced by solving the equation system of two lines that cross themselves, whereas second step just calculates the position of  $M_R^f$  by proportional distances (see Figure 5). As for third step, the expressions of attraction force  $\vec{F}_{att}$  and repulse force  $\vec{F}_{rep}$  are given by the Evolution of Artificial Potential Field (EAPF) stated by [16]:

$$\vec{F}_{rep} = \begin{cases} \vec{F}_{att} = 2K_a \vec{r}_{goal} \\ -K_r \left( \frac{1}{\rho} - \frac{1}{\rho_0} \right)^2 \left( \frac{\vec{r}_{goal}}{\rho^2} + \frac{1}{2} \hat{n} \right) & \text{if } \rho < \rho_0 \\ 0 & \text{if } \rho \geq \rho_0 \end{cases} \quad (2)$$

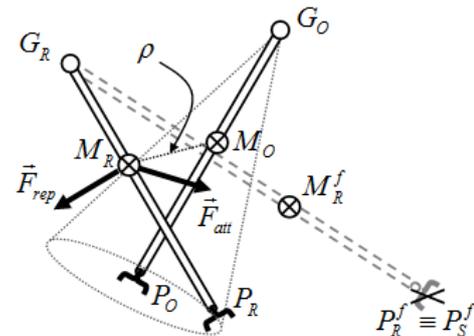


Figure 5 Virtual force for generating trajectories to the target.

Figure 5 shows both forces presented in (2), being  $K_a$  and  $K_r$  their respective gains. Repulsive force  $\vec{F}_{rep}$  only acts on a bound surrounding the obstacle below distance  $\rho_0$  and has two components: one part depends on the distance to the goal  $\vec{r}_{goal}$  as well as the distance to the obstacle  $\rho$ , and the other is a constant with a direction  $\hat{n} = (\hat{r}_{goal} \times \hat{\rho}) \times \hat{\rho}$  perpendicular to  $\hat{\rho}$  and always pointing to the target. As a result, this constant forces the robot to move even if it has reached a local minima situation.

To adapt the potential field of the obstacle tool to a cone a new gain  $K'_r$  may be defined which depends on the relation between the obstacle tool total length  $L$  and the distance from its fulcrum point  $G_O$  and the point of minimal distance with the robot  $M_O$  (see Figure 5):

$$K'_r = \frac{\overline{G_O M_O}}{\overline{G_O P_O}} K_r = \frac{l}{L} K_r \quad (3)$$

Finally, the expression of  $\vec{v}_1$  is deduced in expression (4) from the virtual force  $\vec{F}_1$  of (1), where virtual mass  $m$  has been considered to be 1:

$$\vec{v}_1 = \int \frac{\vec{F}_1}{m} dt \rightarrow \vec{v}_1(t + \Delta t) = \vec{v}_1(t) + \frac{\vec{F}_1}{m} \Delta t \quad (4)$$

#### 4.2 VELOCITY CORRECTIONS

APF behaviour loses its efficiency in dynamic environments because it gives a trajectory just based on the actual state. This way, a fuzzy algorithm is included to take velocity changes and path predictions into account. For example, if the robot moves on the same direction as the obstacle, there will be little or no variation on the velocity calculated by the APF method. However, if the robot moves nearby the obstacle and it moves in a perpendicular way, then it is possible that both future trajectories cross themselves. The fuzzy system will change the robot velocity and try to avoid the collision by reducing the speed in order to allow the surgeon to follow his or her original trajectory [17]. All possible situations are described by the truth table II.

The outputs on this table are the speed corrections  $\vec{v}_2$  previously presented on Figure 4 produced by each combination of antecedent data. These rules adapt the speed of the robot tool so that the obstacle passes before; thus, the cells at the bottom-left of the table, which corresponds to an obstacle that will cross a close point in the trajectory after a long period of time, produce very slow commands. A value of S (Stop) means that  $\vec{v}_2 = -\vec{v}_1$ , whereas MM (Max Movement) indicates that  $\vec{v}_2 = \vec{0}$ , which fits with a maximum velocity generated by the APF behaviour already explained in section 4.1.

Table II - Truth Table for Velocity Corrections

Distance \ Time	Z	AZ	M	F	VF
Z	S	SM	NM	MM	MM
AZ	S	SM	NM	MM	MM
M	SM	SM	NM	FM	MM
L	AS	AS	SM	NM	MM
VL	AS	AS	AS	SM	FM

Z = Zero, AZ = Almost Zero, M = Middle, F = Far, VF = Very Far.

L = Long, VL = Very Long.

S = Stopped, AS = Almost Stopped, SM = Slow Movement, NM = Normal Movement, FM = Fast Movement, MM = Max Movement

#### 4.3 BACKWARD MOVEMENT

With the fuzzy system stated above, almost all situations may be covered. However, collision may also occur if the surgeon moves his or her tool rightly to the robot and forces the contact. This could happen because the surgeon needs to apart for some reason the robot tool for a moment. Thus, the natural solution consists of moving the robot instrument rightly into the obstacle velocity direction. This way, the robot would follow the obstacle as long as it is moving, and should stop when the obstacle does. Only when the obstacle frees the default APF trajectory, the robot would restart its normal auto-navigation.

More specifically, the behaviour proposed will be likely a damp between the obstacle and the robot tool. This way, a virtual force will move the robot back at the obstacle trajectory direction which expression is (5):

$$\vec{F}_3 = B\vec{v}_O \quad (5)$$

The gain  $B$  on (5) is a constant which indicates how fast the robot velocity will react to the obstacle. If this parameter is high, then the robot tool will change its velocity very fast to reach the obstacle velocity  $\vec{v}_O$  fast, and similar conclusions can be taken for low values of  $B$ .

Therefore, the resulting velocity of the robot tool can be obtained by adding the three behaviours exposed (6):

$$\vec{v}_R = c_1\vec{v}_1 + c_2\vec{v}_2 + c_3\vec{v}_3 \quad (6)$$

The parameters  $c_1$ ,  $c_2$  and  $c_3$  on expression (6) are the importance of each velocity  $\vec{v}_1$ ,  $\vec{v}_2$  and  $\vec{v}_3$ , and are estimated by the Fuzzy decision explained in the beginning of this section (see Figure 4).

#### 5 IMPLEMENTATION AND EXPERIMENTS

This section describes the experiments considered to validate the proposed methodology for auto-guided movements. For this purpose, it has been used the PA-10

manipulator system from Mitsubishi Heavy Industries, Ltd., which can be viewed on Figure 6. In this picture, the manipulator has a force sensor attached to give information about the fulcrum location and to allow interaction with the surgeon as well as the patient [14]. To complete the implantation, the optical tracker Polaris Spectra from NDI gives information on the surgeon's tools location. This sensor recovers data on surgeon's position and orientation for both tools simultaneously.

Once the physical system is described, this work proposes to take gauze to the surgeon's tool location. Therefore, the experiment goal is to find a trajectory in real time between the robot initial location and its target by avoiding the obstacle tool. The robot tool is supposed to be already inside the abdomen with gauze on its tool tip. Two situations are considered: one with no movement on the obstacle tool and another one with a dynamic behaviour of the surgeon.

### 5.1 STATIC SURGEON

As it can be shown in Figure 7, the situation where the obstacle does not move is solved by the APF behavior. Gauze is carried by the robot at its tool tip. The robot tool trajectory represented by asterisks is linear until it feels the obstacle tool potential field. Once the robot reaches this zone, it changes the trajectory by surrounding the obstacle at a certain distance (denoted  $\rho_0$  in section 4.1). When the robot bypasses the obstacle zone, it continues the linear trajectory until it reaches the target.

Figure 7 also shows the velocity commanded to the robot. The APF velocity appears with dashed line, whereas the commanded one is solid. It can be noted that the module of the robot velocity is the maximum possible when the tool is far from the obstacle, but it shrinks when the distance between the robot tool and the obstacle is closer. The velocity finally increases again when the robot is near enough of the target.

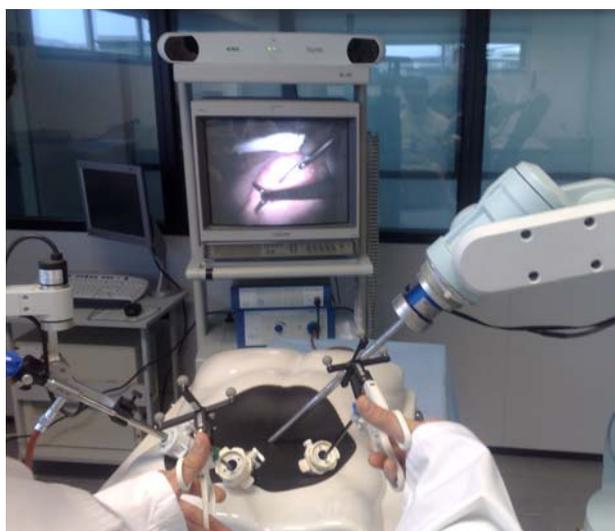


Figure 6 Experimental setup with the auto-guided algorithm proposed on this work.

### 5.2 DYNAMIC SURGEON

Figure 8 shows the resulting trajectory of the robot tool with an asterisk line when the obstacle has freedom of movement. The obstacle trajectory, drawn as a series of dots over the obstacle tool, has been obtained by optical tracking measurements. It can be noticed that the robot not only displaces due to its proximity to the obstacle, but also changes its direction and speed. This way, the resulting trajectory has some zones where the robot has to adapt its trajectory because the obstacle tool is very close to the robot instrument.

The velocity behaviour of the robot tool tip is pretty similar to the static experiment, as shown in Figure 8. This time, the velocity of the obstacle is also represented by a dotted line. The main difference with the static obstacle appears when the robot tool is very close to the obstacle. It can be noted that the velocity in this situation is higher than the one planned by the APF behaviour, because the robot must follow the obstacle velocity.

## 6 CONCLUSION

This work is a first step on auto-guided movements for autonomous assistance to the surgeon. Developers have found a valid strategy for the robot assistant to find free-obstacle paths to a target location. In order to validate the methodology, a robotic arm system has been used for implementation purposes. An experiment where the robot should take gauze to the surgeon's tool has been developed with success.

However, as future works the author believes that a deeper interaction between the camera movements and the robot tool would improve auto-guiding tasks. Also, there is another important issue not covered in this paper: the robot-patient interaction. It is important to generate safe trajectories for the robot avoiding tissue collisions. There may be also tasks where the robot must directly interact with the patient.

## REFERENCES

- [1] Kumar R., Jensen P., Taylor R.H., Experiments with a Steady Hand Robot in Constrained Compliant Motion and Path Following. *IEEE International Workshop on Robot and Human Interaction*, 1999.
- [2] Hynes P., Dodds G.I., Wilkinson A.J., Uncalibrated visual-servoing of a dual-arm robot for surgical tasks. *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, 2005.
- [3] Krupa A., Gangloff J., Doignon C., Mathelin M.F., Morel G., Leroy J., Soler, L., Marescaux J., Autonomous 3-D positioning of surgical instruments in robotized laparoscopic surgery using visual servoing. *IEEE Transactions on Robotics and Automation*, 2003.

[4]

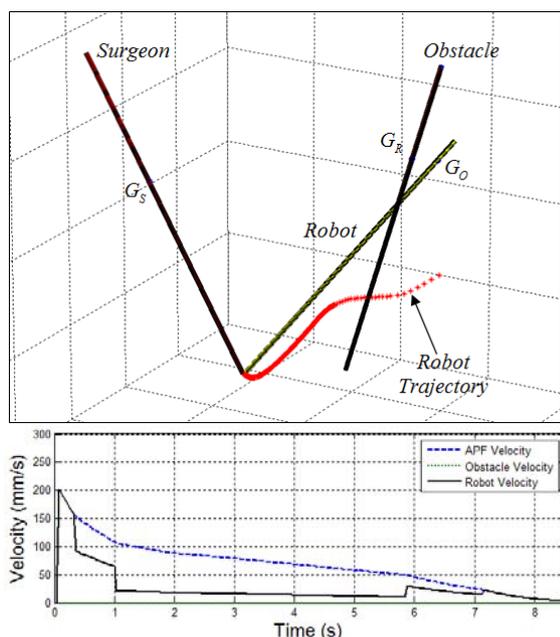


Figure 7 Robot trajectory when the surgeon's obstacle tool remains static (above) and resulting velocity modules (below).

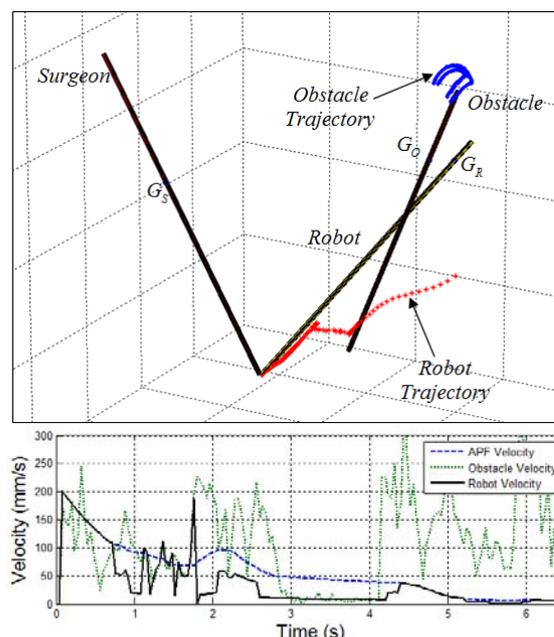


Figure 8 Robot trajectory when the surgeon's obstacle tool moves (above) and resulting velocity modules (below).

- [5] Gangloff J., Ginhoux R., Mathelin M., Soler L., Marescaux J., Model predictive control for compensation of cyclic organ motions in teleoperated laparoscopic surgery. *IEEE Transactions on Control Systems Technology*, 2006.
- [6] Ortmaier T., Gröger M., Boehm D.H., Falk V., Hirzinger G., Motion Estimation in Beating Heart Surgery. *IEEE Transactions on Biomedical Engineering*, 2005.
- [7] Kang H., Wen J.T., Robotic assistants aid surgeons during minimally invasive procedures. *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 1, 2001.
- [8] Chen G., Pham M.T., Redarce T., A semi-autonomous micro-robotic system for Colonoscopy. *IEEE International Conference on Robotics and Biomimetics*, 2008.
- [9] Ho-seok S.; Ki-young K.; Jung-ju L., Development of the dexterous manipulator and the force sensor for Minimally Invasive Surgery. *International Conference on Autonomous Robots and Agents*, 2009.
- [10] Dumpert J., Lehman A.C., Wood N.A., Oleynikov D., Farritor S.M., Semi-autonomous surgical tasks using a miniature in vivo surgical robot. *IEEE International Conference on Engineering in Medicine and Biology Society*, 2009.
- [11] Mayer H., Nagy I., Knoll A., Schirmbeck E.U., Bauernschmitt R., The Endo[PA]R system for minimally invasive robotic surgery. *IEEE International Conference on Intelligent Robots and Systems*, 2004.
- [12] Mayer H., Nagy I., Burschka D., Knoll A., Braun E.U., Lange R., Bauernschmitt R., Automation of Manual Tasks for Minimally Invasive Surgery. *International Conference on Autonomic and Autonomous Systems*, 2008.
- [13] Brett P.N., Taylor R.P., Proops D., Coulson C., Reid A., Griffiths M.V., A surgical robot for cochleostomy. *IEEE International Conference of the Engineering in Medicine and Biology Society*, 2007.
- [14] Bauzano, E., Munoz V.F., Garcia-Morales I., Estebanez B., Three-layer control for active wrists in robotized laparoscopic surgery. *IEEE International Conference on Intelligent Robots and Systems*, 2009.
- [15] Irajli R., Manzuri-Shalmanit M. T., A New Fuzzy-Based Spatial Model for Robot Navigation among Dynamic Obstacles. *IEEE International Conference on Control and Automation*, 2007.
- [16] Enxiu S., Tao C., Changlin H., Enxiu S., Junjie G., Study of the New Method for Improving Artificial Potential Field in Mobile Robot Obstacle Avoidance. *IEEE International Conference on Automation and Logistics*, 2007.
- [17] Fernandez R., Mandow A., Munoz V.F., Garcia-Cerezo A., Real-Time Motion Control for Safe Navigation. *IFAC Symposium on Intelligent Autonomous Vehicles*, 1998.



# QUADROTOR CONTROL BASED ON PARTIAL SENSOR DATA

Laszlo Kis, Bela Lantos

Department of Control Engineering and Information Technology  
Budapest University of Technology and Economics

## ABSTRACT

A miniature quadrotor helicopter is an underactuated nonlinear system. Usually several types of sensors are used to control it. This article focuses on the emergency situation, when only few sensors are available. These sensors are mainly the inertial ones. Certainly, these sensors are not enough for normal manoeuvring, but the controlled system should be able for horizontal stabilization and safe landing. Two types of control are presented. The first is a casual LQ state feedback. The disadvantage of this type of control in real environment is shown. The second control algorithm is an  $H_\infty$  control method. The control design starts from a very simple linear model. The parameter measurement of the linear system is shown. The goal of this article is to handle the uncertainty in the behaviour of the real system.

Keywords: Quadrotor helicopter, state estimation, helicopter control, robust control

## 1 INTRODUCTION

Unmanned aerial vehicles are important types of autonomous vehicles. Helicopters have the advantage that they can move in six degree of freedom with low speed. Mechanically, a robust type is the quadrotor design. This type has been forgotten for a long time, because in small size, they have fast unstable behaviour. Nowadays, there are controllers and sensors, which are fast enough to stabilize a quadrotor helicopter.

### 1.1 RELATED WORK

In the last few years many research group started to work with quadrotors. Most of them concentrate on a specific field of quadrotor design. For example [1], [2], [3], [4] describe different types of control algorithms. Usually these solutions are tested on a skeleton of a quadrotor helicopter. Others describe the electrical and mechanical solutions of their work. In these cases a complex helicopter is built [5], [6]. Some researchers designed a fully autonomous system, like the STARMAC and RAVEN platform [7], [8]. Quadrotor helicopters appeared in commerce, for example Draganfly and Microdrone products. These units are human controlled, but have built in function for stabilization or position hold.

### 1.2 RECENT PROJECT

Our project started in 2006, in cooperation with the Department of Control Engineering and Information Technology in the Budapest University of Technology and Economics and the System and Control Lab of the Computer and Automation Research Institute of the Hungarian Academy of Sciences. The main goal of the project is to design and build an autonomous indoor quadrotor helicopter.

In this project a vision system and inertial measurements system are used as sensory system. The control and state estimation algorithms of the normal usage are described in [9]. This article focuses on the situation, when some of the sensors are unreachable. This situation can happen for many reasons. For example the vision system cannot estimate the position of the helicopter because of occlusion. Or the radio channel can be too noisy between the vision system and the helicopter.

In these situations the only sensors which can be used are the inertial ones. They can produce only relative information, therefore they can be used only for a short time. The types of manoeuvres are also limited. In these emergency cases, the main object is to hold the helicopter in horizontal orientation to prevent falling. A second objective can be the vertical landing, if the batteries are low.

Solutions in this article can also be used in outdoor situations, where the absolute position and orientation sensing comes from the GPS. In these cases the reliability of the GPS can vary during time.

---

Contact author: Laszlo Kis, Bela Lantos

Department of Control Engineering and Information Technology, Budapest University of Technology and Economics, H-1117 Budapest Magyar Tudosok krt. 2.  
E-mail: {lkis,lantos}@iit.bme.hu

After introduction, the used helicopter model is described in Section 2. Section 3 shows how the parameters were measured. Section 4 introduces the problems and solutions of state estimation. Section 5 describes the control design in normal operation. Section 6 presents the electrical components of the helicopter. Section 7 gives some hints how to operate in emergency situations. In Section 8 a basic LQ feedback result is presented. Finally in Section 9 an  $H_\infty$  control is shown, which is robust against the model uncertainty.

## 2 THE HELICOPTER MODEL

The graph of the coordinate systems of the helicopter can be seen in Figure 1.  $K_W$  is the world frame and  $K_H$  is the frame of the helicopter.  $K_{S,0}$  is the original frame of the inertial measurement unit (IMU). This frame is rotated during the calibration method of the IMU to  $K_S$ . The transformation between  $K_H$  and  $K_S$  is determined by the mechanics of the helicopter. Frames  $K_C$  and  $K_{virt}$  are used by the vision system described in [9].

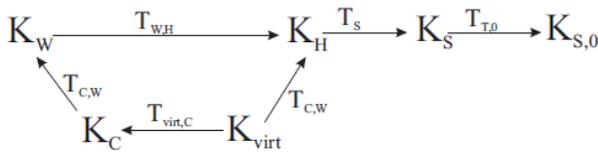


Figure 1 Graph of the frames.

The schematic physical principle of the helicopter is shown in Figure 2.

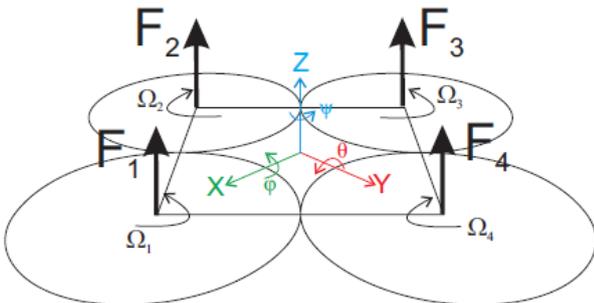


Figure 2 Forces of the quadrotor.

The lift force of one propeller can be calculated according to:

$$F = \Omega^2 b \quad (1)$$

where  $\Omega$  is the revolution of the propeller and  $b$  is a propeller constant. Let  $l$  be the distance between the origin of helicopter's frame and the propeller's axis. Then the torques and lift force can be calculated:

$$\tau_x = (\Omega_4^2 - \Omega_2^2)bl \quad (2)$$

$$\tau_y = (\Omega_3^2 - \Omega_1^2)bl \quad (3)$$

$$\tau_z = (\Omega_2^2 - \Omega_1^2 + \Omega_4^2 - \Omega_3^2)d \quad (4)$$

$$U = \sum_{i=1}^4 b\Omega_i^2 \quad (5)$$

where  $d$  is an other propeller constant.

### 2.1 MOTOR MODEL

The dynamics of the motors should be considered during the modeling process. In our case the motor revolution is controlled by a low-level PI controller, and the closed loop behavior can be approximated with a first order system:

$$\dot{\Omega}_{real} = \frac{1}{T_m}(\Omega_{desired} - \Omega_{real}) \quad (6)$$

where  $T_m$  is the motor time constant. In the followings the indices  $r$  and  $d$  will refer to real and desired value.

Based on (6) and (2)-(5) the dynamic of the real torques and lift forces can be developed as a second order system, like:

$$\tau_d = T_m^2 \ddot{\tau}_r + 2T_m \dot{\tau}_r + \tau_r \quad (7)$$

For an easier control synthesis (7) will be approximated with a first order system:

$$\dot{\tau}_r = \frac{1}{2T_m}(\tau_d - \tau_r) \quad (8)$$

This approximation can be done, if  $T_m$  is small, in our model its value is 0.1s.

### 2.2 ORIENTATION MODEL

Based on the Newtonian law the linear dynamic model for the orientation can be described as

$$\begin{pmatrix} \ddot{\varphi} \\ \ddot{\theta} \\ \ddot{\psi} \end{pmatrix}^T = \begin{pmatrix} \frac{\tau_{x,r}}{\Theta_x} & \frac{\tau_{y,r}}{\Theta_y} & \frac{\tau_{z,r}}{\Theta_z} \end{pmatrix}^T \quad (9)$$

where  $\Theta$  is the inertia around the axes of the coordinate system.

The equations are valid in the frame of the helicopter. These equations remain acceptable in the world frame, if the orientation of the helicopter remains horizontal. Therefore these can be considered as a linearized model around  $\eta = 0$ , where  $\eta$  is the 3D Euler(RPY) orientation of the helicopter relative to world frame.

In this dynamic model, the effect of the aerodynamic friction and the gyroscopic effect considered to be negligible. The aerodynamic friction is in linear connection

with  $\dot{\eta}$  which is approximated with zero. The gyroscopic effect is a function of the rotor inertia  $\Theta_r$ , this can be ignored, because the mass of each rotor is 0.4% of the full mass of the helicopter and this value is also in connection with the difference between the rotor's revolution, which is around zero in a horizontal orientation.

### 2.3 POSITION MODEL

Based on also the Newtonian law the position states are in the following form:

$$\ddot{x}_W = \frac{(C_\varphi S_g C_\psi + S_\varphi S_\psi) U_r}{m} \quad (10)$$

$$\ddot{y}_W = \frac{(C_\varphi S_g S_\psi - S_\varphi C_\psi) U_r}{m} \quad (11)$$

$$\ddot{z}_W = -g + \frac{C_\varphi C_g}{m} U_r \quad (12)$$

where  $C_\bullet$  and  $S_\bullet$  refers to cosine and sine and  $g$  is the gravity acceleration.

### 3 PARAMETER MEASUREMENT

The unknown parameters are the  $b$  and  $d$  parameters of the propellers and the  $l$  and  $\Theta$  parameters of the helicopter.

The parameter  $l$  is the simplest, because this distance can be measured even with a measuring tape. In the case of our helicopter its value is 25 cm.

The parameter  $b$  was determined by using lift force measurement. Lift force was measured in different rotor velocities and a parabola was fitted by using LS method. In our case the value of  $b$  is  $1.4525 \cdot 10^{-5} N s^2$ .

The  $d$  parameter is in connection with the torque around the vertical axis. Even if this value is in connection with  $l$ , it is easier to handle as a standalone parameter. It can be measured with the fixation of all the six degree of freedom, except the rotation around the vertical axis. The torque around the axis should be measured for several rotor velocities and can be approximated by a parabola. In our measurement it is one order less than  $b$ .

The inertia matrix can be determined with a torsion pendulum. The scheme of the pendulum can be seen in Figure 3.

Inertia matrix is approximated as a diagonal one based on the symmetric shape of the helicopter. The inertia around one axis can be determined by the following formula:

$$\Theta = \frac{T_0^2 mgr^2}{4\pi^2 l_{pend}} \quad (13)$$

where  $l_{pend}$  is the length of the two torsion thread and  $r$  is the half distance between the threads.  $m$  is the mass of the helicopter and  $g$  is the gravity constant.  $T_0$  is the period

time of the pendulum. This value was measured with a time-stamped camera.

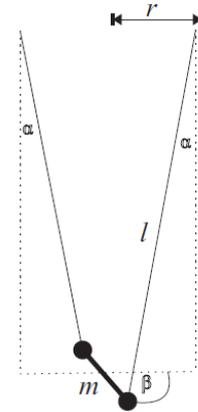


Figure 3 Torsion pendulum.

### 4 STATE ESTIMATION

For control algorithms, position and orientation information is needed. Moreover a sensor fusion between the IMU and the vision system should be performed.

These problems are solved with extended Kalman filters. In first stage the orientation is estimated. Then based on orientation information the position is estimated.

#### 4.1 ORIENTATION ESTIMATION

For orientation estimation the sensor model is the following:

$$\omega_S = \omega_{S,real} + \omega_{S,b} + \omega_{S,n} \quad (14)$$

$$\omega_{H,real} = A_s \omega_S - A_s \omega_{S,b} + A_s \omega_{S,n} \quad (15)$$

where  $A_s$  is the rotation part of  $T_s$  and indices  $b$  and  $n$  refer to bias and noise, respectively. The transformation between  $\omega_H$  and  $\dot{\eta}$  is

$$F = \begin{bmatrix} 1 & S_\varphi T_g & C_\varphi T_g \\ 0 & C_g & -S_\varphi \\ 0 & S_\varphi / C_g & C_\varphi / C_g \end{bmatrix} \quad (16)$$

Then the model used for Kalman filter is

$$\dot{\eta} = FA_s \omega_S - FA_s \omega_{S,b} + FA_s \omega_{S,n} \quad (17)$$

$$\dot{\omega}_{S,b} = \omega_{S,b,n} \quad (18)$$

$$\dot{\eta}_v = \eta + \eta_n \quad (19)$$

where  $\eta_v$  is the orientation information from the vision system. With Euler approximation for state  $x = (\eta, \omega_{S,b})^T$ ,  $u = \omega_S$  and  $y = \eta_v$ , a discrete time system can be formulated:

$$x_{k+1} = f(x_k, u_k, w_k) \quad (20)$$

$$y_k = g(x_k, z_k) \quad (21)$$

where  $w$  and  $z$  are state and measurement noises. With the following definition:

$$R_{w,k} = E[w_k w_k^T], R_{z,k} = E[z_k z_k^T] \quad (22)$$

$$A_k = \frac{\partial f(\hat{x}_k, u_k, 0)}{\partial x}, B_{w,k} = \frac{\partial f(\hat{x}_k, u_k, 0)}{\partial w} \quad (23)$$

$$C_{k+1} = \frac{\partial g(\bar{x}_{k+1}, 0)}{\partial x}, C_{z,k+1} = \frac{\partial g(\bar{x}_{k+1}, 0)}{\partial z} \quad (24)$$

The extended Kalman filter has the form:

$$\bar{x}_k = f(\hat{x}_{k-1}, u_{k-1}, 0) \quad (25)$$

$$M_k = A_{k-1} \Sigma_{k-1} A_{k-1}^T + B_{v,k-1} R_{v,k-1} B_{v,k-1}^T \quad (26)$$

$$S_k = C_k M_k C_k^T + C_{z,k} R_{z,k} C_{z,k}^T \quad (27)$$

$$G_k = M_k C_k^T S_k^{-1} \quad (28)$$

$$\hat{x}_k = \bar{x}_k + G_k (y_k - g(\bar{x}_k, 0)) \quad (29)$$

$$\Sigma_k = M_k - G_k S_k G_k^T \quad (30)$$

In the case when there is no new information from the vision system, in (29) simply  $\hat{x}_k = \bar{x}_k$  to be applied.

#### 4.2 POSITION ESTIMATION

The model for position estimation is the following:

$$\dot{v} = -\omega \times v + A_S (a_S - a_{S,b} + a_{S,n}) - \varepsilon \times \rho - \omega \times (\omega \times \rho) + A_\eta^{-1} g \quad (31)$$

$$\dot{a}_{S,b} = a_{S,b,n} \quad (32)$$

$$\dot{p} = A_\eta v + v_{p,n} \quad (33)$$

$$p_m = p + p_n \quad (34)$$

where  $a$ ,  $v$  refers to acceleration and velocity in  $K_H$ ,  $p$  is the position in  $K_W$ ,  $\varepsilon$  is the angular acceleration (obtained with the numeric derivation of  $\omega$ ),  $\rho$  is the position part of  $T_s$  and  $A_\eta$  is the Euler (RPY) rotation based on the angles  $\eta$ . From this system an extended Kalman filter can be formulated similarly to the orientation case.

#### 5 NORMAL CONTROL (ALTITUDE)

The control algorithms are based on the continuous state space model (9)-(12). Controllers are implemented in discrete time, hence the model was transformed with first order Euler approximation.

The Linear-Quadratic design minimizes the following error statement:

$$J = \sum_{k=0}^{\infty} (x_k^T Q x_k + u_k^T R u_k) \quad (35)$$

where  $Q$  and  $R$  are positive definite matrices.

The weight matrix of the state vector is  $Q$ . The goal for the control is the output converges to the desired value. Therefore the highest value in  $Q$  should correspond to the output state.

With the  $R$  matrix the speed of the control can be set. On the other hand a faster control means higher steps in actuator signals. A reliable compromise between  $Q$  and  $R$  should be found.

The pure linear altitude subsystem is:

$$\frac{d}{dt} (\dot{z}_w \ z_w \ U_r)^T = \left( \frac{C_\varphi C_g}{m} U_r \ \dot{z}_w \ \frac{1}{2T_m} (U_d - U_r) \right)^T \quad (36)$$

The difference between (36) and (12) is the  $-g$  offset in  $z_w$ . The LQ feedback can be designed to (36) and in the control  $g$  should be added to  $U_d$ . The behaviour of the real control can be seen in Figure4.

During the normal operation the helicopter can hold its altitude. In this period an average  $U_d$  value can be obtained, which will be useful in emergency control.

The control for the other subsystems can be designed similarly.

#### 6 REAL TIME SOLUTION

The scheme of the realization of the helicopter can be seen in Figure 5.

The main processing unit of the helicopter is a Freescale MPC555 microcomputer equipped with two CAN buses. The first one is the dedicated bus between the motor controllers and the MPC555. It is necessary to maintain a reliable and fast communication channel for the actuator signals. The second CAN bus is for any other communication.

The CAN bus is a priority based communication interface. The most important data on the CAN bus are the sensor signals.

The helicopter has two types of sensory systems. The first one is an mNAV100CA inertial measurement unit which is on board of the helicopter. On CAN bus the information from the angular velocity and acceleration sensors are propagated. The second sensory system is a visual feedback from an external vision system, which is able to calculate the position and the orientation of the helicopter in the world frame around 20 to 30 sample per second. It runs on a host PC and data are sent by an RF Zigbee communication interface. The inertial information has the highest priority on the CAN channel and the vision information has the second highest.

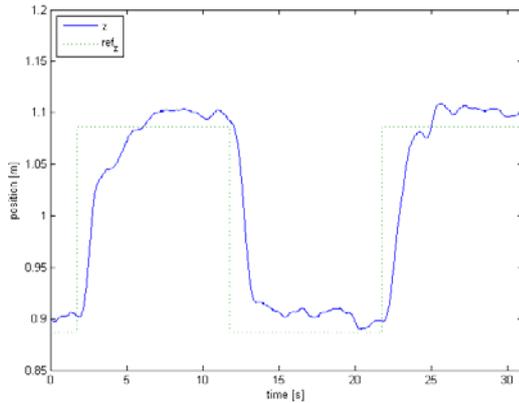


Figure 4 Result of the altitude control.

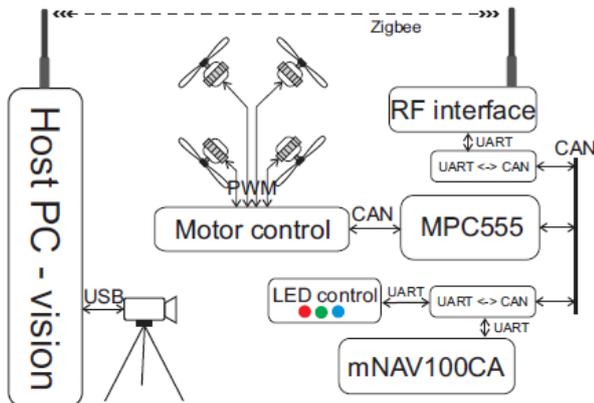


Figure 5 Realization of the helicopter.

Unfortunately both the IMU and the RF module have no CAN interfaces. Therefore the packages should be translated to the CAN format using Atmel AT90CAN128 microcontrollers.

The vision system has an active marker based solution. These markers are LEDs. For the efficiency of the vision unit the colours of the LEDs need to be different. To reach this requirement, RGB LEDs are used. It is rewarding, at least in the prototyping period, if the colours can be changed by software. Therefore each LED is controlled by its own individual controller.

## 7 EMERGENCY CONSIDERATION

In the case of the real time test of this paper, only the angular velocity sensors of the IMU were used in order to emulate emergency situation. To use the measured data a calibration method should be performed. This is described in [9].

The calibration method gives a usable result in the case of extended Kalman filters. However in emergency situation the calculation method of orientation is much more simple, only a numeric integration of the angular velocity data is used.

During the first tests it seemed that the bias is rapidly varying during the flight. This is because of the generated

air flow, which changes the temperature of the sensors. Fortunately there is a thermistor built in each angular velocity sensor chip. This information can be used to produce more accurate results.

The temperature dependence can be measured during the offline calibration, but a varying temperature environment is needed. It is not a complicated condition, because after switch on the system, the temperature of the sensors starts to increase. This change is enough for the calibration.

During the offline calibration the sensors are in stationary position, hence the measured angular velocity is only the biases. Combining with temperature measurement, a temperature-bias characteristic is given, which is approximated with a linear dependency. Then for an other measurement the bias were calculated based on the previous characteristic. The results are in the Figure 6.

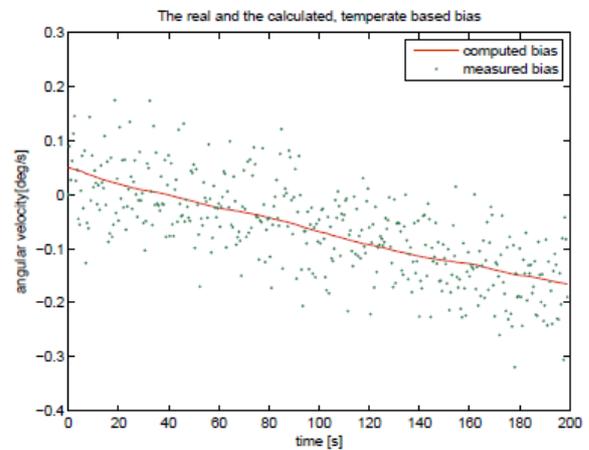


Figure 6 Measured and temperature based calculated bias of an angular velocity component.

The corrections of the measurements are done by the MPC555 microcomputer. This unit solves also the integration of the angular velocity. In the case of angular velocity the component based integration leads to a false result. For proper result the Rodriguez-formula is used. In this case the axis of the rotation is the axis of the measured three dimensional angular velocity vector, and the angle of the rotation is the magnitude of the angular velocity vector multiplied by the sampling time.

$$t_k = \frac{(\dot{\varphi}_k, \dot{\vartheta}_k, \dot{\psi}_k)^T}{\sqrt{\dot{\varphi}_k^2 + \dot{\vartheta}_k^2 + \dot{\psi}_k^2}} \quad (37)$$

$$\alpha_k = T_S \sqrt{\dot{\varphi}_k^2 + \dot{\vartheta}_k^2 + \dot{\psi}_k^2} \quad (38)$$

where  $T_S$  is the sensor sampling time, in this case  $0.01s$ .

The starting rotation matrix is the unit matrix. In each sampling period the actual rotation matrix should be multiplied with the Rodriguez rotation ( $R_R$ ):

$$R_{R,k} = C_{\alpha_k} \cdot I + (1 - C_{\alpha_k})[t_k \circ t_k] + S_{\alpha_k}[t_k \times] \quad (39)$$

$$R_{k+1} = R_{R,k} R_k \quad (40)$$

where  $I$  is the unitary matrix and  $R_k$  describes the actual rotation. The  $(\varphi, \vartheta, \psi)$  orientation angle can be calculated by the solution of the inverse Euler (RPY) problem. Figure 7 shows, how the error accumulates during the orientation calculation process. In the situation the sensors stood in stationary orientation, therefore the expected value is zero.

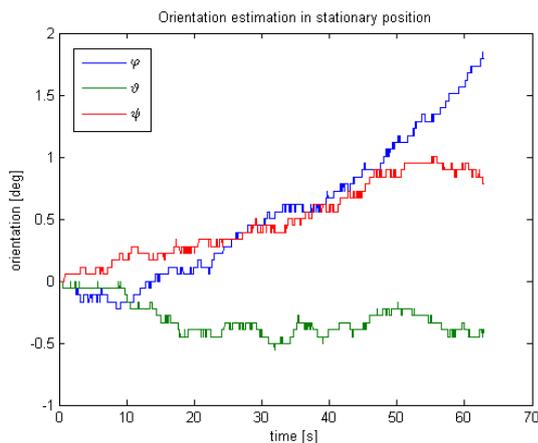


Figure 7 Orientation estimation in stationary state.

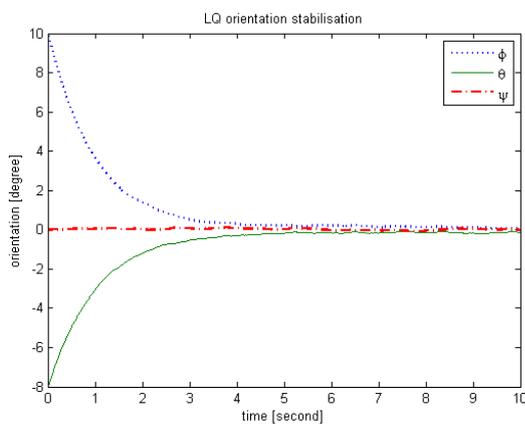


Figure 8 Orientation stabilisation with LQ control (simulation).

## 8 LQ STATE FEEDBACK (ORIENTATION)

In the following chapters only the emergency control is discussed.

The state vector  $x_k$  in (35) contains the values of angular velocity and orientation. In this case the main goal is to stabilize the orientation, therefore the weight of the orientation was set two order higher than the ones of the angular velocity.

## 8.1 SIMULATIONS

The results of the simulation can be seen in Figure 8. The simulation was also done with a more complex nonlinear model, based on [1], but around horizontal orientation the two models are almost equivalent. During simulation the sensor noise is considered.

## 8.2 REAL FLIGHT

The results of a real flight can be seen in Figure 9. This contains a vertical takeoff and landing. The main difference between the simulation and the real test is that in real situation the average of the orientation is not zero. This is because of the inadequate modelling of the real helicopter. Many source of this difference can be found. For example, the mass of the helicopter cannot be concentrated to the origin of the helicopter frame. Or the small differences between the propellers and motors can cause that the value  $b$  in the model differs from those of the motors. One solution for this problem can be the introduction of four different  $b$  parameters and the identification of them. In practice it is hard, because a complex identification process is needed after every change of the helicopter components.

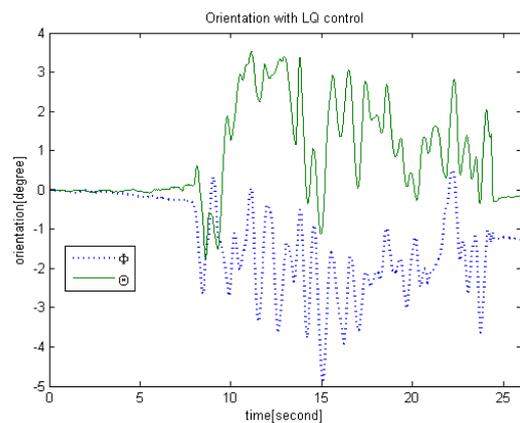


Figure 9 Orientation control with LQ (real test)

## 9 H<sub>∞</sub> STABILIZATION

Another solution for the previous problem can be if the model stays as before, but the control removes the orientation error. This can be solved by using integrator in the controller.

Let the continuous case be investigated. The classical solution [10] is to extend the state space system with an augmented state vector:

$$\dot{x}_I = \int y dt \quad (41)$$

Then the augmented state space system is:

$$\frac{d}{dt} \begin{pmatrix} x \\ x_I \end{pmatrix} = \begin{bmatrix} A & 0 \\ C & 0 \end{bmatrix} \begin{pmatrix} x \\ x_I \end{pmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u \quad (42)$$

$$\begin{pmatrix} y \\ x_I \end{pmatrix} = \begin{bmatrix} C & 0 \\ 0 & I \end{bmatrix} \begin{pmatrix} x \\ x_I \end{pmatrix} \quad (43)$$

The previous LQ design method could be used to this system. However, here a more robust  $H_\infty$  control design is presented. The  $\Delta - P - K$  structure of the  $H_\infty$  synthesis can be seen in Figure 10 [11].

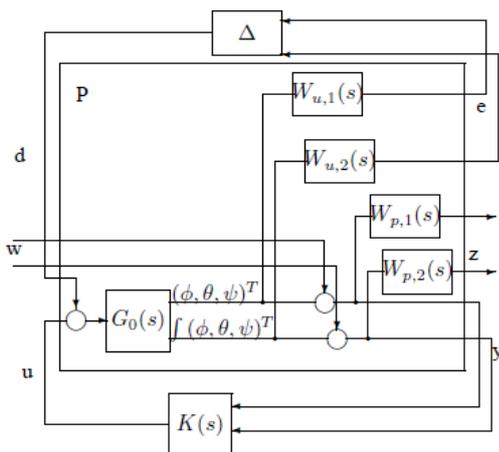


Figure 10 Structure of  $H_\infty$  synthesis.

The weights  $W_u$  and  $W_p$  refers to model uncertainty and performance, respectively. Weights are set in the way, that in high frequency the uncertainty can be high, and in low frequency the stabilization of  $y$  should be fast. The resulted  $K(s)$  controller is converted to discrete time with step response equivalence.

### 9.1 SIMULATION

The result of the stabilization can be seen in Figure 11. In this case the sensor noise is also simulated. It should be noted, that this type of  $H_\infty$  design doesn't remove the whole orientation error, but the remaining error can be set with the weights of the performance outputs to a very small value.

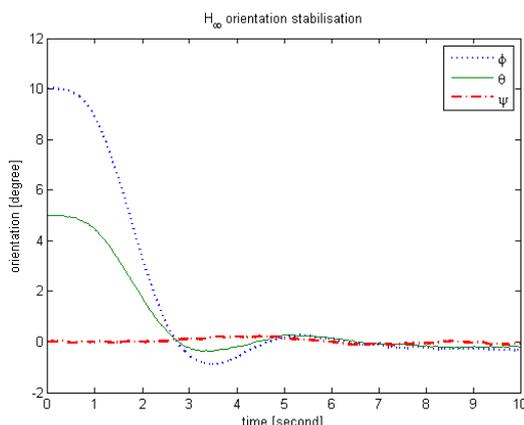


Figure 11  $H_\infty$  stabilization (simulation).

### 9.2 REAL FLIGHT

The result of a complex real take off, flight and landing is shown in Figure 12. The orientation gets much closer to zero than in the case of LQ control. A small error still remains, as it was expected after the simulation.

It should be noted, that the orientation starts from a zero value, but after landing the orientation has a small amount of offset. The bias of the angular velocity sensor is compensated by the temperature, but other components can also change the bias. The remaining error is caused by the integration of the remaining bias.

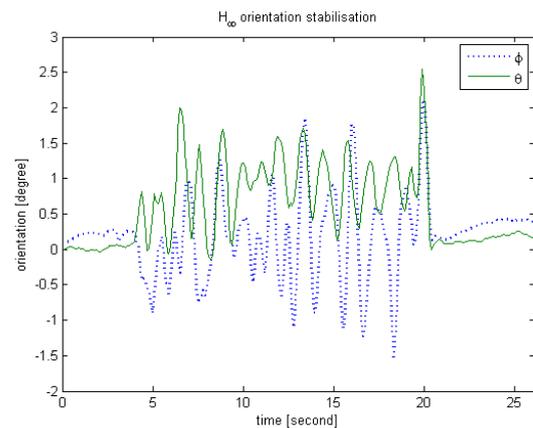


Figure 12  $H_\infty$  stabilization (real test).

## 10 CONCLUSION

In this paper a quadrotor helicopter control is presented for the case where only the inertial sensors are reliable. In our case this situation can happen if the vision system cannot compute the orientation and position of the helicopter. It can be considered an emergency situation, in which the first requirement is to stabilize the orientation of the helicopter.

The presented control can also be used as a startup controller to execute a take off and make a horizontal stabilization. During this period the controller can handle and estimate the uncertainty behaviour, which can be useful in controlling complex manoeuvres. It is also shown, why a simple state feedback controller is not sufficient for quadrotor control.

In the emergency control system only the angular velocity sensor is used. A temperature compensation method is used for improving angular velocity integration.

### 10.1 FUTURE WORK

The goal of our project is to establish a quadrotor system consisting three individual helicopters, which are able to flight in formation. During this way, the next step is to combine the emergency controller presented in this paper with a more complex one [12]. Then the system will be able to fly through a preprogrammed path and perform safe landing also in emergency situation.

The second future plan is to combine a quadrotor helicopter with a differential GPS system and perform outdoor operations.

## ACKNOWLEDGMENT

This work is connected to the scientific program of the "Development of quality-oriented and harmonized R+D+I strategy and functional model at BME" project. This project is supported by the New Hungary Development Plan (Project ID: TÁMOP-4.2.1/B-09/1/KMR-2010-0002) The research was also supported by the Hungarian National Research Program under grant No. OTKA K 71762.

## REFERENCES

- [1] Bouabdallah S., Murrieri P. and Siegwart R., Design and Control of an Indoor Micro Quadrotor. *International Conference on Robotics and Automation*, New Orleans, USA, 2004.
- [2] Madani T., Benallegue A., Control of a Quadrotor Mini-Helicopter via Full State Backstepping Technique. *IEEE Conference on Decision and Control*, San Diego, USA, 2006.
- [3] Das A., Lewis F., Subbarao K., Backstepping Approach for Controlling a Quadrotor Using Lagrange Form Dynamics. *Journal of Intelligent and Robotic Systems*, September 2009, Vol. 56, pp. 127-151.
- [4] Voos H., Nonlinear control of a quadrotor micro-UAV using feedback-linearization. *IEEE International Conference on Mechatronics*, Malaga, 2009.
- [5] Hanford S. D., Long L. N. and Horn J. F., A Small Semi Autonomous Rotary-Wing Unmanned Air Vehicle (UAV). *American Institute of Aeronautics and Astronautics, Infotech@Aerospace Conference*, Paper No. 2005-7077, 2005.
- [6] Pounds P., Mahony R., Hynes P. and Roberts J., Design of a Four-Rotor Aerial Robot. *Australasian Conference on Robotics and Automation*, 2002.
- [7] Hoffmann G., Rajnarayan D. G., Waslander S. L., Dostal D., Jang J. S. and Tomlin C. J., *The Stanford Testbed of Autonomous Rotorcraft for Multi Agent Control (Starmac)*. Digital Avionics Systems Conference, 2004.
- [8] How J.P., Bethke B., Frank A., Dale D. and Vian J. Real-time indoor autonomous vehicle test environment, *IEEE Control Systems Magazine*, 28(2) pp. 51-64. ISSN: 0272-1708, 2008.
- [9] Kis L., Prohaszka Z. and Regula G., Calibration and Testing Issues of the Vision, Inertial Measurement and Control System of an Autonomous Indoor Quadrotor Helicopter, *International Workshop on Robotics in Alpe-Adria-Danube Region*, Ancona, Italy, 2008.
- [10] Khoo B. C., Golnaraghi F., *Automatic Control Systems*, Wiley, ISBN 0-471-13476-7
- [11] Zhou K., Doyle J.C., Glover K., *Robust and Optimal Control*, Prentice Hall, Englewood Cliffs, New Jersey 07632
- [12] Kis L., Regula G. and Lantos B., Design and Hardware-in-the-Loop Test of the Embedded Control System of an Indoor Quadrotor Helicopter, *Workshop on Intelligent Solutions in Embedded Systems*, Regensburg, Germany, 2008.

# SIMULATION INVESTIGATIONS ON A NOVEL APPROACH TO MODEL REFERENCE ADAPTIVE CONTROLLERS

J.K. Tar, I.J. Rudas, J.F.Bitó\*      K.R. Kozłowski\*\*      C. Pozna\*\*\*

\*Institute of Intelligent Engineering Systems, John von Neumann Faculty of Informatics, Óbuda University,  
Bécsi út 96/B., H-1034 Budapest, Hungary, E-mail: {tar.jozsef@nik., rudas@, bito@}uni-obuda.hu

\*\*Chair of Control and Systems Engineering, Computing Science & Management, Poznań University of Technology,  
Piotrowo 3a, 60-965 Poznań, Poland, E-mail: krzysztof.kozlowski@put.poznan.pl

\*\*\*Design and Robotics Department, University Transilvania of Brasov,  
Bd. Eroilor 29. Brasov Romania, E-mail: cp@unitbv.ro

## ABSTRACT

The development of “Model Reference Adaptive Controllers (MRAC)” is a popular approach from the early nineties to our days. Their main point is the application of proper feedback in an internal loop that makes the behavior of the so manipulated system identical to that of the “reference model” that normally is simple enough to be controlled by an external loop. It has many particular variants with the common feature that they normally are designed by the use of Lyapunov's 2nd (“direct”) method with a quadratic Lyapunov function constructed of the tracking error and further additional terms. Though this approach normally guarantees global asymptotic stability, its use can entail complicated tuning that may have disadvantages whenever very fast applications are needed. In this paper the operation of an alternative solution using “Robust Fixed Point Transformations (RFPT)” in the design phase is investigated via simulations for a 3 degree of freedom system. This approach applies strongly saturated, multiplicative nonlinear terms causing a kind of “deformation” of the input of the available imprecise system model within a local basin of convergence. The stability of this controller is proved by Lyapunov's direct method.

Keywords: Model Reference Adaptive Control, Lyapunov's Direct Method, Robust Fixed Point Transformations

## 1 INTRODUCTION

The MRAC technique is a popular and efficient approach in the adaptive control of nonlinear systems e.g. in robotics. A great manifold of appropriate papers can be found for the application of MRAC from the early nineties (e.g. [1] to our days (e.g. [7]). One of its early applications was a breakthrough in adaptive control. In [3] C. Nguyen presented the implementation of a joint-space adaptive control scheme that was used for the control of a noncompliant motion of a Stewart platform-based manipulator that was used in the Hardware Real-Time Emulator developed at Goddard Space Flight Center to emulate space operations. It comprised two platforms and six linear actuators driven by DC motors, and possessed six degrees of freedom. Experimental studies have shown that the adaptive control scheme provided superior tracking capability in comparison with fixed-gain controllers.

The mainstream of the adaptive control literature at that time used some parametric models and applied Lyapunov's “direct method” for model parameter tuning (e.g. [1], [2]). This tuning method normally entails some difficulties and deficiencies. Certain deficiencies and potential improvements of the most fundamental adaptive controllers designed for robotics as the “Slotine--Li Adaptive Controller” and the “Adaptive Inverse Dynamics Controller” [1] were discussed/introduced in [10] and [11] that essentially consisted in dropping the use of the Lyapunov function in parameter tuning. These approaches still guaranteed global asymptotic stability that was shown by the use of certain terms that were the “fragments” of the original Lyapunov function, but the tuning rule was far faster and contained far less number of almost arbitrary parameters than that of the positive definite matrices of the conventional Lyapunov functions. However, the existence

of external disturbances unknown by the controller could “fob” or “mislead” the tuning process of these controllers. Moreover, it was observed that the faster the tuning process was the more significant the effect of the external disturbances was [12].

The MRAC technique normally tunes certain control parameters that are not parts of the dynamic model of the system to be controlled. It was successfully applied e.g. for impedance [5] control, motion control [6] of manipulator arms as well as for teleoperation purposes [7]. It was also combined with the use of “Artificial Neural Networks” [8] and fractional order systems [9].

The above mentioned approaches had the common feature that they applied Lyapunov functions for control and parameter tuning. Normally it is a difficult task to find appropriate Lyapunov function and guarantee its non-increasing nature that leads to complicated estimations while finding the appropriate tuning rule.

To overcome the complexity of Lyapunov function based techniques alternative approaches are also present in the literature. For instance, as a model-based technique the idea of “Situational Control” can be mentioned in connection with the control of turbojet engines [13]. It is related to the observation that “... *in practice the turbojet engine finds itself in very different working conditions that influence parameters of its operation and characteristics of behavior. By creation of algorithms of control, it is necessary to create models in the whole dynamic spectrum of the modeled system (turbojet engine) and also its erroneous states.*”. In [13] the Authors focused “*on implementation methods of situational control as a framework method, which is suitable for use in design of dynamic models and systems of control of turbojet engines with use of intelligent elements that comply with full authority control digital systems standards.*” Another possibility is the application of robust switching controllers that were successfully applied for crane models and swinging reduction [14].

For avoiding the use of Lyapunov's technique the potential application of iterative fixed point transformations with local basin of attraction of convergence was studied for the adaptive control of “smooth” physical systems e.g. in [15]. The robustness of this approach later was improved in [16]. These controllers worked on the basis of the concept of “Complete Stability” that is widely used in connection with the operation of the Cellular Neural Networks, too [17]. This method applied the concept of the “expected - realized system response” and was found to be appropriate to also developing an MRAC technique for “Single Input - Single Output (SISO)” systems [18]. The aim of the present paper is to show that the variant of these fixed point transformations developed for MIMO systems also is useful for revisiting and simplifying the design of MRAC controllers. In the sequel at first the idea of the “expected - realized system response” and the Robust Fixed Point Transformations are briefed. In contrast to the previous approaches a Lyapunov function will be used for showing

the stability of the recommended method. However, this Lyapunov function is not needed in the phase of designing the particular controller. Following that the application of this method for MRAC purposes is analyzed. Finally simulation results are presented for a 6 degree of freedom cart+beam+hammer system and the available conclusions are drawn. The present paper is a modified and extended version of the brief conference report in [19]. The essence of the modifications and extension can be summarized as follows: a) instead of the technique of Cauchy sequences a Lyapunov function based technique is used for showing the convergence of the method; b) the originally used “hand-made” preliminary SCLAB program with simple Euler integration with identical time-resolution and discrete control steps has been replaced by a SCILAB-SCICOS program using its far more sophisticated ODE solver that automatically selects integration method depending on the stiffness of the problem and also allows carefully limiting the allowable maximum of the discrete time-steps that allows the approximation with “continuous differential equations” instead “discrete controllers”; c) the well behaving 3rd order spline functions in the definition of the nominal trajectories to be tracked have been replaced by a trajectory also having drastically accelerating/decelerating segments.

## 2 ADAPTIVE CONTROL ON THE BASIS OF THE EXPECTED REALIZED SYSTEM RESPONSE

Certain control tasks can be formulated by using the concept of the appropriate “excitation”  $U$  of the controlled system to which it is expected to respond by some “desired response”  $x^d$ . The appropriate excitation can be computed by the use of the inverse dynamic model of the system as  $U=\varphi(x^d)$ . Since normally this inverse model is neither complete nor exact, the actual response determined by the system's dynamics,  $\psi$ , results in a realized response  $x^r=\psi(\varphi(x^d))=f(x^d)\neq x^d$  that differs from the desired one. It is worth noting that the functions  $\varphi()$  and  $\psi()$  may contain various hidden parameters that partly correspond to the dynamic model of the system, and partly pertain to unknown external dynamic forces acting on it. The controller normally can manipulate or “deform” the input value from  $x_d$  to  $x^*$  so that  $x^d=\psi(\varphi(x^*))=f(x^*)$ . Such a situation can be maintained by the use of some local deformation that can properly “drag” the system's state in time while it meanders along some trajectory. To realize this idea a fixed point transformation was applied in [16] that is quite “robust” as far as the dependence of the resulting function on the behavior of  $f()$  is concerned. This robustness can approximately be investigated by the use of an affine approximation of  $f(x)$  in the vicinity of  $x^*$  and it is the consequence of the strong nonlinear saturation of the sigmoid function  $\tanh(x)$  in the definition (1):

$$H(x, f(x) - x^d) := (x + K) \left[ 1 + B \tanh(A[f(x) - x^d]) \right] - K \quad (1)$$

If  $f(x_*) - x^d = 0$  then  $H(x_*, 0) = x_*$   
For arbitrary  $x$ :  $H(-K, f(x) - x^d) = -K$

It is evident that the transformation  $H$  defined in (1) has a proper ( $x^*$ ) and a false ( $-K$ ) fixed point, according to the 2nd line of the group (1). We note that instead of the function  $\tanh()$  any sigmoid function, i.e. any bounded, monotone increasing, smooth function  $\sigma(x)$  with the property of  $\sigma(0)=0$  can be used for our purposes. Let us use (1) for a 2nd order SISO systems in the following manner. Let  $q^N(t)$  denote the *nominal trajectory to be tracked by the controlled system*, and let  $q(t)$  denote the *actual one* that is realized by the controller. For prescribing a PID-type tracking error decrease in purely kinematic terms the variable  $\xi(t) := \int_0^t [q^N(\zeta) - q(\zeta)] d\zeta$  (the integrated tracking

error assuming that the controller commences its operation at  $t=0$ ) and a  $\Lambda > 0$  constant can be introduced to define the “desired joint coordinate acceleration”  $\ddot{q}^d$  for which

$$\left( \frac{d}{dt} + \Lambda \right)^3 \xi(t) = 0. \text{ It evidently yields}$$

$$\ddot{q}^d(t) \equiv h(t) := \ddot{q}^N(t) + \Lambda^3 \xi(t) + 3\Lambda^2 \dot{\xi}(t) + 3\Lambda \ddot{\xi}(t). \quad (2)$$

If (2) is exactly satisfied then evidently the tracking error  $e(t) := q^N(t) - q(t) \rightarrow 0$  as well as  $\xi \rightarrow 0$ . Let us introduce a fixed delay time  $\tau > 0$ , and by observing the system's behavior “in the past” (i.e. in the instant  $t-\tau$ ) apply the “deformed input”  $z$  at instant  $t$  as  $z(t) := H(z(t-\tau), f(z(t-\tau), q(t-\tau), \dot{q}(t-\tau)) - h(t))$ . The 2nd order system's response at time  $t$  will be  $\ddot{q}(t) = f(z(t), q(t), \dot{q}(t))$  since it can be assumed that e.g. in the case of a Classical Mechanical System's actual physical state is determined by the generalized variables  $(q(t), \dot{q}(t))$  that also influence its state propagation  $(\dot{q}(t), \ddot{q}(t))$ . Utilizing the fact that  $H$  is smooth and assuming that the system is in the vicinity of the fixed point of  $H$  it can be written that

$$\begin{aligned} z(t) &= H(z(t-\tau), f(z(t-\tau), q(t-\tau), \dot{q}(t-\tau)) - h(t)) \approx \\ &\approx H(z(t-\tau), 0) + \partial_2 H \cdot [f(z(t-\tau), q(t-\tau), \dot{q}(t-\tau)) - h(t)] = \\ &= z(t-\tau) + \partial_2 H(z(t-\tau), 0) \cdot [f(z(t-\tau), q(t-\tau), \dot{q}(t-\tau)) - h(t)] \end{aligned} \quad (3)$$

in which  $\partial_2 H$  denotes the partial derivative of  $H$  according to its 2nd argument and it has to be calculated at the arguments  $(z(t-\tau), 0)$ . Assuming that  $f$  is a smooth function (that can be achieved by properly defining the available rough system model) and that the system is under smooth influences, for small  $\tau$  it can be written that

$$\begin{aligned} f(z(t), q(t), \dot{q}(t)) &\approx f(z(t-\tau), q(t-\tau), \dot{q}(t-\tau)) + \\ &+ \partial_1 f \cdot [z(t) - z(t-\tau)] + \partial_2 f \cdot [q(t) - q(t-\tau)] + \\ &+ \partial_3 f \cdot [\dot{q}(t) - \dot{q}(t-\tau)] \end{aligned} \quad (4)$$

( $\partial_i f$  denotes the partial derivative of  $f$  according to its  $i$ th argument, and these derivatives can be taken at time  $t-\tau$ ). Equation (4) evidently means that

$$\begin{aligned} \ddot{q}(t) &\approx \ddot{q}(t-\tau) + \partial_1 f \cdot [z(t) - z(t-\tau)] + \\ &+ \partial_2 f \cdot [q(t) - q(t-\tau)] + \partial_3 f \cdot [\dot{q}(t) - \dot{q}(t-\tau)]. \end{aligned} \quad (5)$$

By substituting  $z(t) - z(t-\tau)$  from (3) into (5) we obtain that

$$\begin{aligned} \ddot{q}(t) &\approx \ddot{q}(t-\tau) + \partial_1 f \cdot \partial_2 H(z(t-\tau), 0) \times \\ &\times [f(z(t-\tau), q(t-\tau), \dot{q}(t-\tau)) - h(t)] + \\ &+ \partial_2 f \cdot [q(t) - q(t-\tau)] + \partial_3 f \cdot [\dot{q}(t) - \dot{q}(t-\tau)] = \\ &= \ddot{q}(t-\tau) + \partial_1 f \cdot \partial_2 H(z(t-\tau), 0) \times \\ &\times [q(t-\tau) - h(t)] + \partial_2 f \cdot [q(t) - q(t-\tau)] + \partial_3 f \cdot [\dot{q}(t) - \dot{q}(t-\tau)] = \\ &= \ddot{q}(t-\tau) + \partial_1 f \cdot \partial_2 H \times \\ &\times [\ddot{q}(t-\tau) - h(t-\tau) - h(t) + h(t-\tau)] + \\ &+ \partial_2 f \cdot [q(t) - q(t-\tau)] + \partial_3 f \cdot [\dot{q}(t) - \dot{q}(t-\tau)] \end{aligned} \quad (6)$$

Via subtracting  $h(t) = h(t-\tau) + h(t) - h(t-\tau)$  from both sides we obtain that

$$\begin{aligned} \ddot{q}(t) - h(t) &\approx \ddot{q}(t-\tau) - h(t-\tau) - [h(t) - h(t-\tau)] + \\ &+ \partial_1 f \partial_2 H \cdot [\ddot{q}(t-\tau) - h(t-\tau) - \{h(t) - h(t-\tau)\}] + \\ &+ \partial_2 f \cdot [q(t) - q(t-\tau)] + \partial_3 f \cdot [\dot{q}(t) - \dot{q}(t-\tau)] = \\ &= [\ddot{q}(t-\tau) - h(t-\tau)] (1 + \partial_1 f \partial_2 H) - \\ &- \{h(t) - h(t-\tau)\} (1 + \partial_1 f \partial_2 H) + \\ &+ \partial_2 f [q(t) - q(t-\tau)] + \partial_3 f [\dot{q}(t) - \dot{q}(t-\tau)] \end{aligned} \quad (7)$$

By dividing both sides of this equation by the small delay  $\tau$  the finite element approximation of a differential equation can be obtained for the joint acceleration error  $\zeta(t) := \ddot{q}(t) - h(t)$  as

$$\begin{aligned} \dot{\zeta}(t-\tau) &\approx \frac{\zeta(t) - \zeta(t-\tau)}{\tau} \approx \zeta(t-\tau) \frac{\partial_1 f \partial_2 H(z(t-\tau), 0)}{\tau} - \\ &- \dot{h}(t-\tau) (1 + \partial_1 f \partial_2 H) + \partial_2 f \dot{q}(t-\tau) + \partial_3 f \dot{\dot{q}}(t-\tau) \end{aligned} \quad (8)$$

From (1) it is easy to see that

$$\begin{aligned} \partial_2 H(x, f(x) - x^d) &= (x + K) \left[ BA \tanh(A[f(x) - x^d]) \right] \\ \partial_2 H(x, 0) &= (x + K) BA \end{aligned} \quad (9)$$

If we have information on the sign of  $\partial_i f$  on the basis of some analytical argumentation then the setting  $B = -\text{sign}(\partial_i f)$  can be chosen. By developing a low quality simple traditional PID controller simulation investigations can be made for observing the absolute maximum value of the occurring responses so a value  $K_0 \gg \max|\ddot{q}|$  can be

chosen. (It essentially depends on the accelerations of the nominal trajectory and the feedback constants applied.) The “dynamics” of the whole system determines the reasonable choice for the delay time  $\tau > 0$  which is *short enough for not allowing the “past observations” become obsolete during it*. With this choice we obtain that (8) approximately becomes

$$\begin{aligned} \zeta(t-\tau) \approx & \frac{-K_0 A |\partial_1 f|}{\tau} \zeta(t-\tau) - \dot{h}(t-\tau) \partial_1 f \partial_2 H + \\ & + \partial_2 f \ddot{q}(t-\tau) + \partial_3 f \dot{q}(t-\tau) \end{aligned} \quad (10)$$

Following that, on the basis of simple simulations the  $A > 0$  can so be chosen that we obtain a damping factor for  $\zeta(t)$  as  $\alpha := \frac{K_0 A |\partial_1 f|}{\tau} > 0$  that exponentially stabilizes it around 0 according to (10). If  $\alpha$  is large enough the additional terms at the RHS of (10) mean only little perturbation. Now, for defining a Lyapunov function at first a formal artificial state variable  $\mathbf{y} := [\zeta, \dot{\zeta}, \ddot{\zeta}]^T$  must be defined that propagates according to the equation

$$\dot{\mathbf{y}} = \begin{bmatrix} \dot{\zeta} \\ \ddot{\zeta} \\ \ddot{\zeta} \end{bmatrix} = \begin{bmatrix} \zeta \\ \dot{\zeta} \\ \ddot{\zeta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -\Lambda^3 & -3\Lambda^2 & -3\Lambda \end{bmatrix} \begin{bmatrix} \zeta \\ \dot{\zeta} \\ \ddot{\zeta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ h - \dot{q} \end{bmatrix} \quad (11)$$

With a positive definite constant matrix  $\mathbf{P}$  of size  $3 \times 3$  the Lyapunov function  $V := \mathbf{y}^T \mathbf{P} \mathbf{y}$  can be introduced that evidently satisfies the restrictions set for a Lyapunov function [ $V(\mathbf{y}) \geq 0$ ,  $V(\mathbf{0}) = 0$ , and  $V > 0$  if  $\mathbf{y} \neq \mathbf{0}$ ]. Furthermore, according to Barbalat’s lemma from  $\mathbf{y} = \mathbf{0}$  it follows that  $q(t) \rightarrow q^N(t)$ . For guaranteeing non-positive derivative to  $V$ , by the use of (11) the usual condition can be gained:

$$\dot{V} = \mathbf{y}^T (\mathbf{C}^T \mathbf{P} + \mathbf{P} \mathbf{C}) \mathbf{y} + 2 \mathbf{y}^T \mathbf{P} \begin{bmatrix} 0 \\ 0 \\ -\zeta \end{bmatrix} \leq 0. \quad (12)$$

By solving the Lyapunov equation  $\mathbf{C}^T \mathbf{P} + \mathbf{P} \mathbf{C} = -\mathbf{Q}$  for a positive definite symmetric  $\mathbf{Q}$  the quadratic term in (12) becomes negative, while the remaining term is only linear in  $\mathbf{y}$ , and it is multiplied with the quantity  $\zeta$  that fluctuates near zero. Therefore, though no asymptotic stability can be granted, but a simple stability with some little positive  $V$  certainly can be guaranteed until the solution is in the vicinity of the fixed point of  $H(x, 0)$ . The basin of convergence depends on the properties of  $f(x)$  and it can be manipulated by the approximate model data and structure. A possibility for applying the same idea of adaptivity is the application of a sigmoid function projected to the direction of the response-error as e.g. in [20] with  $w(t) := f(t-\tau) - x^d(t)$ ,  $u(t) := w(t) / \|w(t)\|$ ,

$$\mathbf{z}(t) := (1 + B \sigma(A \|\mathbf{w}\|)) \mathbf{z}(t-\tau) + BK \sigma(A \|\mathbf{w}\|) \mathbf{u}. \quad (13)$$

This extension yielded very good simulation results. In the next section the application of this method in the formal framework of MRAC controllers is described.

### 3 APPLICATION OF THE FIXED POINT TRANSFORMATIONS IN MRAC

The above outlined idea can simply be applied for designing MRAC controllers. Let the “desired joint acceleration”  $\ddot{q}^D$  be defined as in the previous section. Let the reference model be used for calculating the desired control action  $U^D$  that could properly govern the reference model. The above outlined input deformation now can simply be applied in the space of the control actions as outlined in Fig. 1 (the block called “Deformation” contains the function  $H$ ). The deformed control signal  $U^{Req}$  is exerted on the actual physical system under control (denoted by the block “System”). Its realized response  $\ddot{q}$  now is introduced into the reference model again, and the appropriate control action calculated from this model is considered to be the realized response in the space of the control actions.

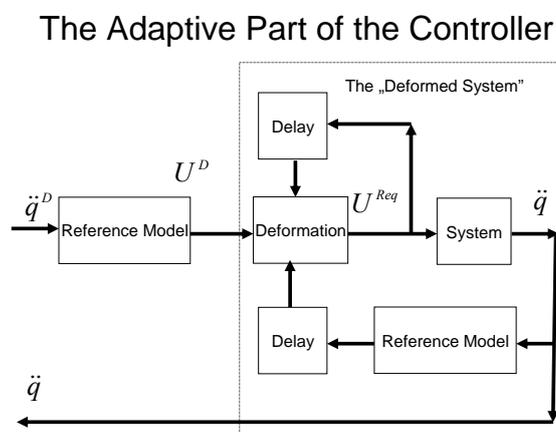


Figure 1 The novel MRAC scheme.

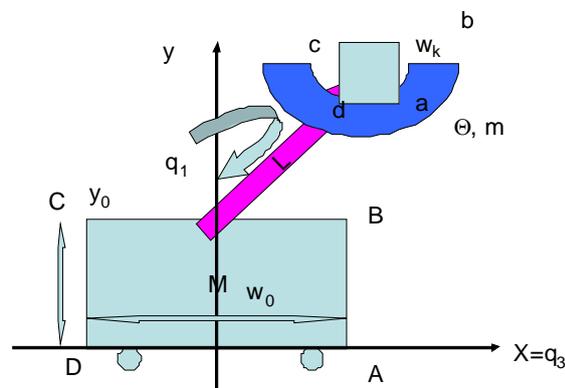
(In the simulations these quantities are referred to as the “Recalculated” control actions.) The scheme in Fig. 1 does not need any further sophisticated mathematical analysis. If it works it evidently has to result in precise trajectory, velocity, and acceleration tracking, and also determines the appropriate deformation of the force / torque signal calculated to the reference model to achieve appropriate acceleration of the actual system under control. Furthermore, as the *recalculated control action* approaches the *desired control action*  $U^D$ , the MRAC idea is also

realized: for the external loop that calculates  $\ddot{q}^D$  the illusion is generated by the internal loop framed in Fig. 1 that it controls a system that dynamically behaves like the reference model. Therefore in the sequel potential application examples are given.

#### 4 SIMULATION EXAMPLES

The model of the paradigm under consideration is depicted in details in Fig. 2.

### Paradigm for Simulation Investigations: Cart + Beam + Hamper System



$$\begin{bmatrix} (mL^2 + \Theta) & \Theta & mL \cos q_1 \\ \Theta & \Theta & 0 \\ mL \cos q_1 & 0 & (m + M) \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} + \begin{bmatrix} -mgL \sin q_1 \\ 0 \\ -mL \sin q_1 \dot{q}_1^2 \end{bmatrix} = \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \end{bmatrix}$$

Figure 2 The dynamic model of the paradigm under consideration.

The exact parameters of the system were  $M=30$  kg,  $m=10$  kg,  $L=2$  m,  $\Theta=20$  kg $\times$ m<sup>2</sup>,  $g=10$  m/s<sup>2</sup>. The approximate model parameters, i.e. the parameters of the reference model were  $\hat{M}=60$  kg,  $\hat{m}=20$  kg,  $\hat{L}=2.5$  m,  $\hat{\Theta}=50$  kg $\times$ m<sup>2</sup>, and  $\hat{g}=8$  m/s<sup>2</sup>. For the PID-type kinematic trajectory tracking in (2) was  $\Lambda=8/s$ . In the adaptive control the  $K=35000$ ,  $B=-1$ ,  $A=0.5 \times 10^{-6}$ , and  $\tau=10^{-4}$  s setting was

applied. The simulations were made in SCILAB-5.1.1 and Scicos version 4.2. In order to get consistent results the maximum allowed time-step of the numerical integration was limited to  $\tau$ .

In the first run results were obtained for a sinusoidal nominal trajectory by the non-adaptive, simple PID controller.

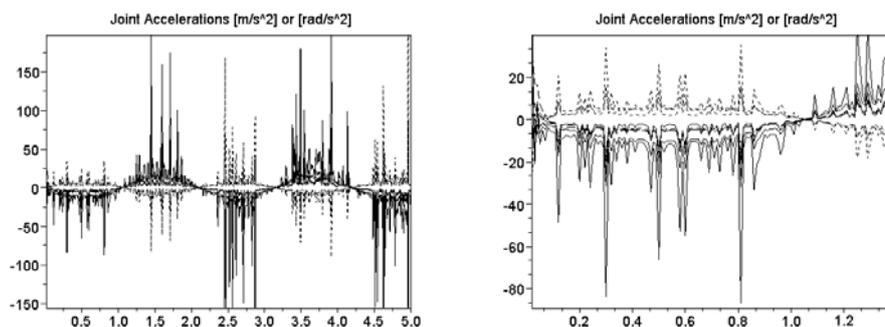


Figure 3 The joint accelerations of the non-adaptive controller tracking a sinusoidal trajectory (for  $q_1$ : solid,  $q_2$ : dashed,  $q_3$ : dash dot, detailed excerpts at the RHS).

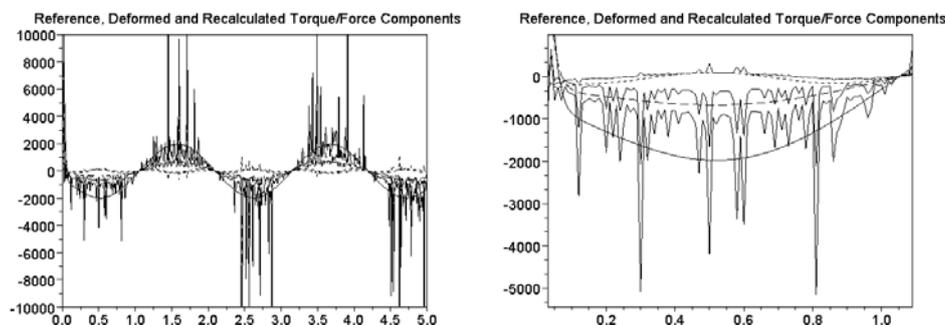


Figure 4 The generalized force components of the non-adaptive controller tracking a sinusoidal trajectory (for  $Q_1$ : solid,  $Q_2$ : dashed,  $Q_3$ : dash dot, detailed excerpts at the RHS).

Figures 3 and 4 reveal that the simple PID controller compensated the modeling errors by very drastic feedback (without adaptivity the “Reference” torque/force signal is exactly identical to the “Deformed” signal because no adaptive deformation happens, so we see only  $3 \times 2$  curves).

It is evident that the “Recalculated” torque/force signal significantly differs from the “Desired” one due to the significant differences between the reference model and the actual system.

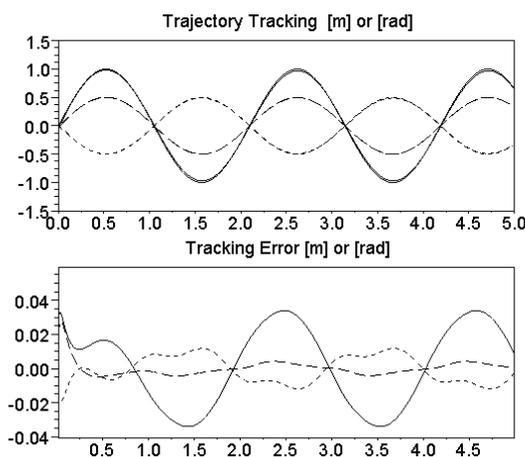


Figure 5 The trajectory tracking of the non-adaptive controller tracing a sinusoidal trajectory (for  $q_1$ : solid,  $q_2$ : dashed,  $q_3$ : dash dot).

Figure 5 reveals that due to the relatively large  $\Lambda$  value the trajectory tracking is not very bad though the MRAC idea naturally is not even approached. The counterparts of the above results for the *adaptive controller* are described in the following figures. Figure 6 reveals that in contrast to the non-adaptive results of Fig. 3 in which the “nominal accelerations”, the “desired accelerations” (i.e. the nominal ones corrected by the PID feedback) and the “simulated/realized accelerations” seriously differ from each other the adaptive controller yields curves in each other’s close vicinity. Furthermore, apart from the initial

transient part that originates from the inappropriate initial velocity of the controlled system the order of magnitude of the joint accelerations are determined by that of the nominal trajectory, so only minimal PID corrections are necessary. In Fig. 7 the realization of the MRAC idea can clearly be observed: for each  $Q_i$  there are two curves in each other’s close vicinity (the “reference” and the “recalculated” values), and their significantly differ from their “deformed” counterpart. The adaptivity of the control is realized through this deformation that also yielded very precise trajectory tracking (Fig. 6).

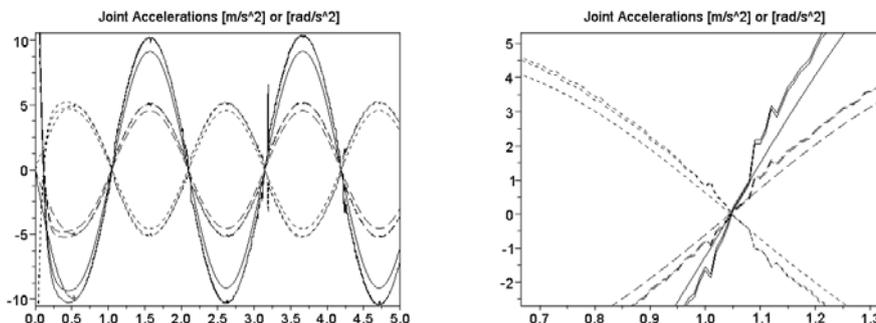


Figure 6 The joint accelerations of the adaptive controller tracking a sinusoidal trajectory (for  $q_1$ : solid,  $q_2$ : dashed,  $q_3$ : dash dot, detailed excerpts at the RHS).

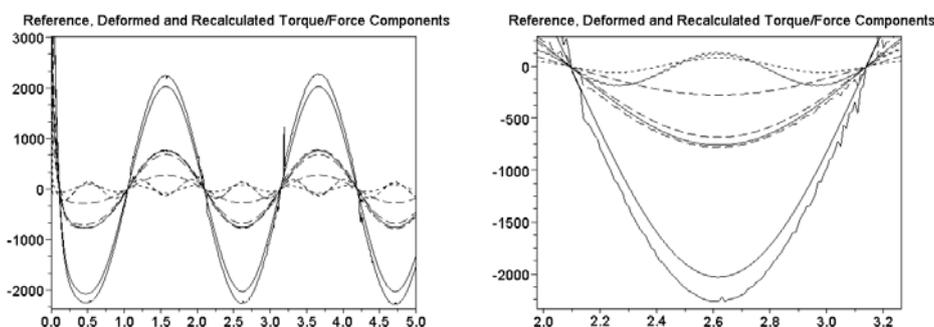


Figure 7 The generalized force components of the adaptive controller tracking a sinusoidal trajectory (for  $Q_1$ : solid,  $Q_2$ : dashed,  $Q_3$ : dash dot, detailed excerpts at the RHS).

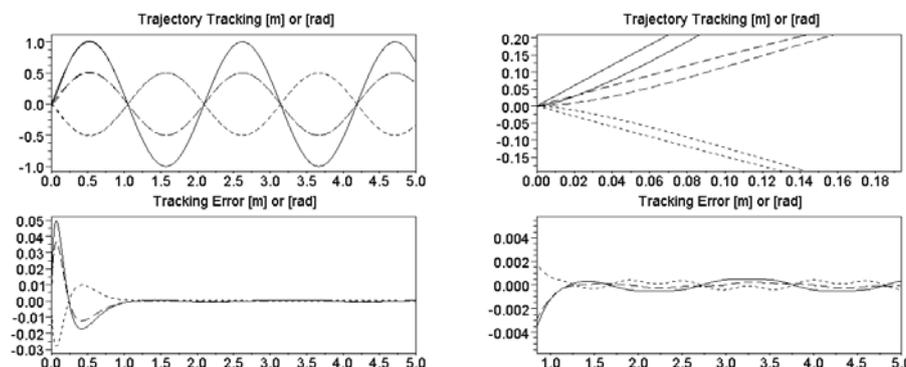


Figure 8 The trajectory tracking of the adaptive controller tracing a sinusoidal trajectory (for  $q_1$ : solid,  $q_2$ : dashed,  $q_3$ : dash dot, detailed excerpts at the RHS).

To further test the abilities of this controller very drastic accelerations/decelerations were introduced into the nominal trajectory by replacing the  $\sin(\omega t)$  type curves by  $\sin^2(\omega t)$  type curves that yields strong deceleration and

subsequent acceleration at  $\sin(\omega t) = 0$ . These new results strictly confirm the observations made for the case of the sinusoidal trajectory.

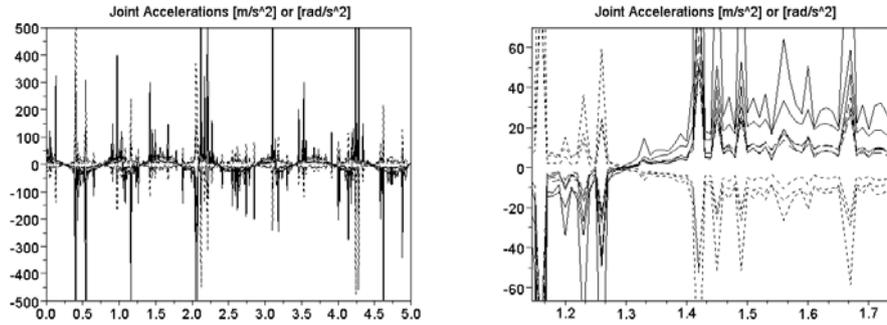


Figure 9 The joint accelerations of the non-adaptive controller tracking a  $\sin^2(\omega t)$  type trajectory (for  $q_1$ : solid,  $q_2$ : dashed,  $q_3$ : dash dot, detailed excerpts at the RHS).

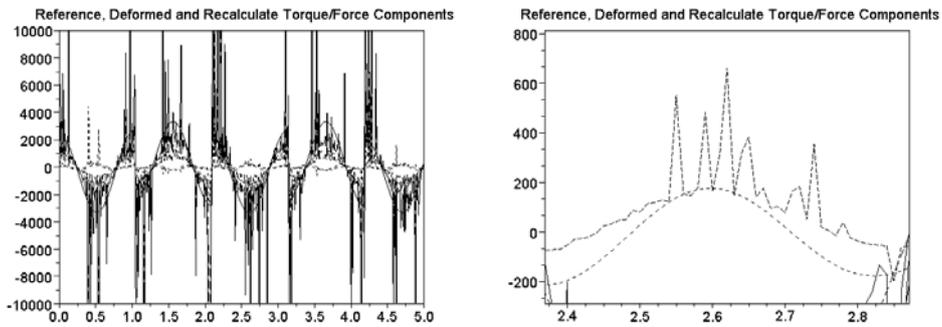


Figure 10 The generalized force components of the non-adaptive controller tracking a  $\sin^2(\omega t)$  type trajectory (for  $Q_1$ : solid,  $Q_2$ : dashed,  $Q_3$ : dash dot, detailed excerpts at the RHS).

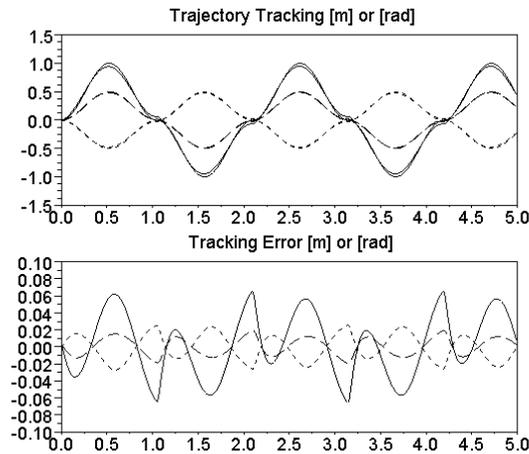


Figure 11 The trajectory tracking of the non-adaptive controller tracing a  $\sin^2(\omega t)$  type trajectory (for  $q_1$ : solid,  $q_2$ : dashed,  $q_3$ : dash dot)

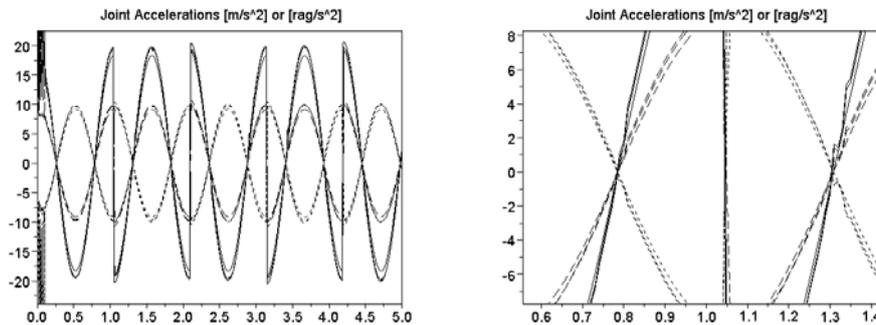


Figure 12 The joint accelerations of the adaptive controller tracking a  $\sin^2(\omega t)$  type trajectory (for  $q_1$ : solid,  $q_2$ : dashed,  $q_3$ : dash dot, detailed excerpts at the RHS).

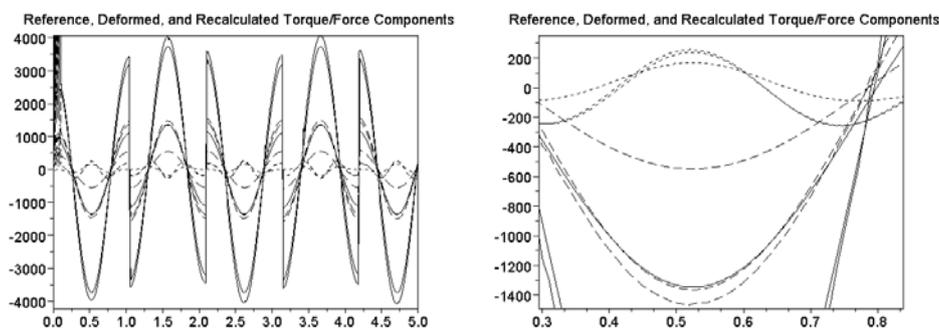


Figure 13 The generalized force components of the adaptive controller tracking a  $\sin^2(\omega t)$  type trajectory (for  $Q_1$ : solid,  $Q_2$ : dashed,  $Q_3$ : dash dot, detailed excerpts at the RHS).

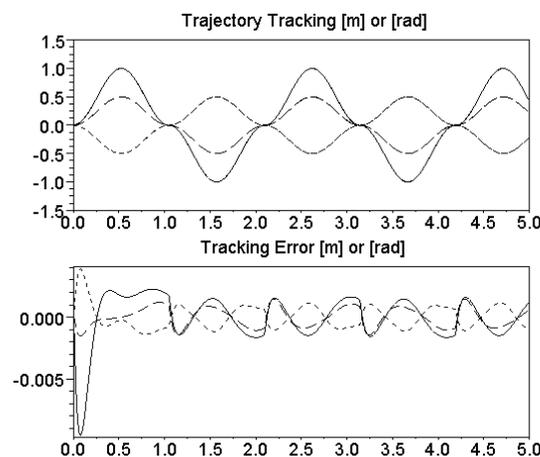


Figure 14 The trajectory tracking of the adaptive controller tracing a  $\sin^2(\omega t)$  type trajectory (for  $q_1$ : solid,  $q_2$ : dashed,  $q_3$ : dash dot).

## 5 CONCLUSIONS

In this paper a novel approach was proposed for realizing the idea of the MRAC controllers for MIMO systems by the use of “Robust Fixed Point Transformations (RFPT)”. While the traditional designs apply Lyapunov’s direct method that normally results in vulnerable and complicated

tuning of many control parameters, this novel one operates with three fixed adaptive control parameters without any tuning. Its main advantage is that it requires only little computational efforts. *In this paper Lyapunov’s direct method was used only for proving the convergence of the novel method. The controller’s design does not need this complicated technique.*

The operation of the method was demonstrated by the example of controlling a 3 DOF Classical Mechanical system with parameter modeling errors. The reference model corresponded to the system having the “nominal model parameters”. The task of the MRAC controller was to make the behavior of the actual system similar to that having the nominal parameters. For the simulations *SCILAB-5.1.1* and *Scicos version 4.2.* were used with a sophisticated ODE solver and properly limited size of the maximum allowable time-step in the numerical integration. A further step in the research may be the investigation of the effects of observation noises especially in the responses of the controlled systems.

#### ACKNOWLEDGMENT

The authors gratefully acknowledge the support by the “*Research and Technology Innovation Fund*” within the project *OTKA: No. CNK-78168.*

#### REFERENCES

- [1] Slotine Jean-Jacques E., Li W., *Applied Nonlinear Control*, Prentice Hall International, Inc., Englewood Cliffs, New Jersey, 1991.
- [2] Isermann R., Lachmann K.H., and Matko D., *Adaptive Control Systems*, Prentice-Hall, New York, 1992.
- [3] Nguyen Charles C., Antrazi Sami S., Zhou Zhen-Lei, Campbell Jr. Charles E., Adaptive control of a stewart platform-based manipulator, *Journal of Robotic Systems*, Vol. 10, No. 5, pp. 657-687, 1993.
- [4] Murray R.M., Li Z., Sastry S.S., *A mathematical introduction to robotic manipulation*, CRC Press, New York, 1994.
- [5] Kamnik R., Matko D. and Bajd T., Application of Model Reference Adaptive Control to Industrial Robot Impedance Control, *Journal of Intelligent and Robotic Systems*, Vol. 22, pp. 153-163, 1998.
- [6] Somló J., Lantos B., Cát P.T., *Advanced robot control*, Akadémiai Kiadó, Budapest, Hungary, 2002.
- [7] Hosseini-Suny K., Momeni H. and Janabi-Sharifi F., Model Reference Adaptive Control Design for a Teleoperation System with Output Prediction, *J Intell Robot Syst*, DOI 10.1007/s10846-010-9400-4, pp. 1-21, 2010.
- [8] Khoh C.J. and Tan K. K., Adaptive robust control for servo manipulators, *Neural Comput & Applic*, Vol. 12, pp. 178-184, 2005.
- [9] Ladaci S. and Charef A., On Fractional Adaptive Control, *Nonlinear Dynamics*, Vol. 43, pp. 365-378, 2006.
- [10] Tar J.K., Rudas I.J., Gáti J., Improvements of the Adaptive Slotine & Li Controller - Comparative Analysis with Solutions Using Local Robust Fixed Point Transformations. *Proc. of the 14th WSEAS International Conference on APPLIED MATHEMATICS (MATH'09)*, Puerto De La Cruz, Canary Islands, Spain, December 14-16, 2009, pp. 305-311, 2009.
- [11] Tar J.K., Bitó, J.F. Rudas, I.J., Preitl S. and Pecup R.-E., An SVD Based Modification of the Adaptive Inverse Dynamics Controller. *Proc. of 5th International Symposium on Applied Computational Intelligence and Informatics*, Timișoara, Romania, 2009, pp. 193-198, 2009.
- [12] Tar J.K., Rudas I.J., Hermann Gy., Bitó J.F., Tenreiro Machado J.A., On the Robustness of the Slotine-Li and the FPT/SVD-based Adaptive Controllers, *WSEAS Transactions on Systems and Control*, Vol. 3, No. 9, pp. 686-700, 2009.
- [13] Andoga R., Föző L., Madarász L., Digital Electronic Control of a Small Turbojet Engine MPM 20, *Acta Polytechnica Hungarica*, Vol. 4, No. 4, pp. 83-95, 2007.
- [14] Hičár M., Ritók J., Robust Crane Control, *Acta Polytechnica Hungarica*, Vol. 3, No. 2, pp. 91-101, 2006.
- [15] Tar J.K., Rudas I.J. and Kozłowski K.R., Fixed Point Transformations-Based Approach in Adaptive Control of Smooth Systems, in *Lecture Notes in Control and Information Sciences 360* (Eds.: M. Thoma and M. Morari), *Robot Motion and Control 2007* (Ed.: Krzysztof R. Kozłowski), Springer Verlag London Ltd., pp. 157-166, 2007.
- [16] Tar J.K., Bitó J.F., Nádai L., Tenreiro Machado J.A., Robust Fixed Point Transformations in Adaptive Control Using Local Basin of Attraction, *Acta Polytechnica Hungarica*, Vol. 6, No. 1, pp. 21-37, 2009.
- [17] Roska T., Development of Kilo Real-time Frame Rate TeraOPS Computational Capacity Topographic Microprocessors. *Plenary Lecture at the 10th International Conference on Advanced Robotics (ICAR 2001)*, Budapest, Hungary, August 22-25, 2001.
- [18] Tar J.K., Bitó J.F., Rudas I.J., Replacement of Lyapunov's Direct Method in Model Reference Adaptive Control with Robust Fixed Point Transformations. *Proc. of the 14th IEEE International Conference on Intelligent Engineering Systems 2010*, Las Palmas of Gran Canaria, Spain, May 5-7, pp. 231-235, 2010.
- [19] Tar J.K., Rudas I.J., Bitó J.F., Kozłowski K.R. and Pozna C., A Novel Approach to the Model Reference Adaptive Control of MIMO Systems. *Proc of the 19th International Workshop on Robotics in Alpe-Adria-Danube Region*, Budapest, Hungary, June 23-25, 2010, pp. 31-36, 2010.

# TIME-OPTIMAL MOTION PLANNING FOR ROBOTS

Somló János\*

Molnár József\*\*

\* Óbuda University  
Knowledge Centre of Robotics  
Budapest, Hungary

\*\* Budapest University of Technology and Economics,  
Dept. Mechatronics, Optics and Mechanical Engineering Informatics

## ABSTRACT

At the practical application of robots, the part processing time has a key role. The part processing time is an idea borrowed from manufacturing technology. In robotics it mostly means that the robots tool-centre point should go along some given path and the tool orientation in every point should also have the given values. These two requirements have to be satisfied at the same time. In the present paper we propose a method which provides the motion in every point of the path with the possible maximum velocity. In fact, we divide the path to transient and cruising parts and require the maximum velocities only for this second part. The given motion is called "Time-optimal cruising motion". We demonstrate on an example the simplicity of the approach. Using the parametric method of motion planning, we give the equations for determining time-optimal motions. Not only the translation motions of tool-centre points but, also the orientation motions of tools may be optimally planned. The time-optimal motion planning is also possible for free paths (PTP motions). A general approach for this problem is proposed, too. In the paper, some parts are devoted to the deeper understanding of the optimization problems.

Keywords: Robot motion planning, Path planning, Trajectory planning, Parametric method, Path length, Time-optimal, Cruising motions, Tool-centre point, Orientation changes, PTP motions, Free paths

## 1 INTRODUCTION

Robot motions may be described by the Lagrange's equation

$$\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\dot{\mathbf{q}}, \mathbf{q}) = \boldsymbol{\tau} \quad (1.1)$$

where  $\mathbf{H}(\mathbf{q})$  is the inertia matrix of the robot,  $\mathbf{q}$  is the vector of joint displacements, their components form the joint coordinates,  $\dot{\mathbf{q}}$  and  $\ddot{\mathbf{q}}$  are the joints velocity and acceleration vectors. The function  $\mathbf{h}(\dot{\mathbf{q}}, \mathbf{q}, t)$  is the nonlinear term containing centrifugal, Coriolis, gravitational forces, frictions and also the external forces affecting the robot joints (including the forces (moments) acting at the end-effectors, too),  $\boldsymbol{\tau}$  is the vector of joint torques. The components of  $\boldsymbol{\tau}$  torques (forces) are restricted by the torques characteristics of the driving motors.

The  $\dot{\mathbf{q}} = (\dot{q}_1, \dot{q}_2, \dots, \dot{q}_n)^T$  components of the joint velocity vector are constrained by the possible maximum number of rotations (in time unit) of the motors. As it is well known, the maximums of torques (forces) are the decisive factors to determine optimal (dynamical) processes. As it will be clear from what is detailed below the constraints of joint velocities determine the time-optimal cruising motions.

Formulating an optimization problem (for example: to move the robot end-effector centre-point from one point to another in the space in minimum time), it can be solved by using the mathematical theory of optimal processes: the Pontriagin's maximum principle, or Dynamic programming of R. Bellman, or other methods.

Let us return to the Lagrange's equation. In extended form it is

$$u_i = \sum_{j=1}^n I_{ij} \ddot{q}_j + \sum_{j=1}^n \sum_{k=1}^n C_{ijk} \dot{q}_j \dot{q}_k + \sum_{j=1}^n R_{ij} \dot{q}_j + g_i \quad (1.2)$$

$$u_{i \min} \leq u_i \leq u_{i \max} \quad \text{where } i = 1, 2, \dots, n$$

In (1.2):

$I_{ij}$  are the components of the inertia matrix,  $C_{ijk}$  are coefficients for the Coriolis and Centrifugal forces. These terms are (usually) also nonlinear functions of joint displacements.

$R_{ij}$  is the viscous damping coefficient, and  $g_i$  is the gravitational term,

$u_i$  is the force or torque given by the actuator of the  $i$ -th joint.

In (1.2) the external forces are not indicated (but, when it is needed, they can be included). It is not indicated either that the components of joint velocities vectors are constrained, too.

Solving the optimal control problem, one may have the solution in form

$$\mathbf{u} = \tau = \mathbf{u}_{\text{opt}}(\dot{\mathbf{q}}, \mathbf{q}, t) \quad (1.3)$$

It can be realized in computed torque manner. But in control practice it is desirable to solve the synthesis problem and generate the control signals depending on the error signals.

The error signals are:

$$\varepsilon_i = q_{id} - q_i \quad i=1, 2, \dots, n \quad (1.4)$$

The  $q_{id}$  signals are the desired values (functions) of the joint coordinates.

Their derivatives are:  $\dot{\varepsilon}_i, \ddot{\varepsilon}_i$  etc.

Looking at Equation (1.2) (having in mind that the coefficients are also highly nonlinear) it can be imagined that to solve optimization task is not an easy task. But if the nonlinear effects can be neglected, in principle, for individual robot arms, the well-known optimal “bang-bang” control principles could be applied. As far as we know, it is not very frequently applied in robotics. The reason is: a robot is not an artillery gun, or a spacecraft, or any similar.

In the present paper we will not follow the way outlined in this Introduction. In the second part of the paper the motion planning problems will be specified and analyzed. Also a state-of-the-art summarization is given. The 3rd part outlines the basic results concerning time-optimal cruising motions. In this part the basics of parametric method of motion planning are given, too. In part 4, time-optimal PTP motion is analyzed and solution method is presented. In part 5, realization aspects are outlined. In part 6, the interrelation of path, trajectory planning, and trajectory tracking is analyzed. In part 7, the problems of a very important practical field, the making of corsets for spinal diseases corrections is discussed. In Appendix, a simple

example for time-optimal cruising trajectory planning is given.

## ROBOT MOTION PLANNING

Now, let us return to the rather exact formulation of the robot motion planning problems. The following tasks should be solved:

- Path planning
- Trajectory planning
- Trajectory tracking

### PATH PLANNING

Given a robot and its environment. The task is to plan a path which results in a transition of the end-effector centre point:

- a) from one position to another position;
- b) through a series of positions;
- c) along a continuous path.

During these actions it may also be required that the orientation of the grippers, or working tools attached to the end-effector have the given orientations. Sometimes, the path planning can be approached as a pure geometric problem, but in many cases, the path, trajectory planning and tracking problem are deeply interconnected. In the cases when these levels can be considered separately, for path planning, optimization problems, with geometric criteria, can be formulated. For example, the goal may be to get the shortest path to walk over a series of points, or avoid obstacles, or avoid obstacles by volumetric bodies, etc. The powerful apparatus of computer geometry can be used to great extent to solve these problems.

### TRAJECTORY PLANNING

Given a path to be followed by the working point (end-effector centre-point) of a robot, and the corresponding orientations of tools attached to it. The dynamic characteristics of robot joints are known including the constraints on torques, forces available at the actuators. The limit values of the joints speeds, the limit values of speeds in Cartesian coordinate system are also given. Possibly, the same is given for accelerations.

Complex knowledge is available about the technological process characteristics (requirements, forces, etc.).

The most general and practical requirement is to find the motion giving minimum time for performing the task.

Other goal may be to find the motion requiring minimum energy.

### TRAJECTORY TRACKING

The task of the trajectory tracking, as it was mentioned above, is to plan the control action that guarantees the realization of the desired trajectories with the necessary accuracies.

## 2 MINIMUM-TIME TRAJECTORY PLANNING. STATE-OF-THE-ART.

In the Introduction the minimum time motions were reviewed. Below, more details are given.

The time optimal control problems can be classified into three categories:

- Motion on constrained path between two endpoints;
- Motion in free workspace between two endpoints;
- Motion in a free workspace containing obstacles.

Concerning the robot motion in free workspace, a number of results are available. In Geering, Guzzella, Hepner, Onder (1986), [3] it has been shown that the time optimal controls of motion in free workspace, are regularly that of switching nature. The maximum torques (forces) are switched for accelerating and decelerating in an appropriate manner. A huge number of papers were dealing with different aspects of the above problem. An overview can be found in S. K. Singh (1991). In Singh's paper a general numerical method to the solution of similar optimization tasks was proposed, too. Discretization and the use of non-linear programming method form the essence of this approach.

In many of the application problems the motion is constrained to a given path. Examples include arc welding , milling, grinding, painting, deburring using robots.

Several researchers have addressed the problem of this constrained motion of robots. Recently, it has turned out that the parametric description of the robot motion is one of the most promising way of the investigation of constrained motion. The most detailed outline of this method can be found in K. G. Shin, N.D.McKay (1991)[9].

When using the parametric method, the differential equations characterising the motion of the joints of an n-degree of freedom robot can be transformed to a form where instead of n joint coordinates  $(q_1, q_2, \dots, q_n)$  only the one path parameter  $(\lambda(t))$  is present. The n non-linear, coupled (second order) differential equation of joints motion is transformed to a second order non-linear differential equation formulated for one parameter. Shin, McKay, and others, based on the parametric description, proposed an approach to the solution of the time-optimal control of constrained motion of robots. Shin and McKay also used the parametric description method to the determination of other than the time-optimal motion. An example is the solution of optimal control problem using minimum energy criterion.

When using the method of parametric description, usually, the parameter is the length  $(\lambda(t))$  along the path. In the present paper this approach will be used to a high extent, with the goal of investigating cruising motion rather than investigating dynamics. J. Podurajev and J. Somlo (1993) [10] used a parameter the time derivative of which is proportional to the square root of the entire kinetic energy of the robot mechanism. Using this parameter, the equation of motion becomes extremely simple. This approach made

possible to develop optimal robot control according to energy criterion in a straightforward manner.

Later, in this paper we introduce the time-optimal cruising trajectory planning problem in detail and solve it. Shortly, we speak about cruising motion when a robot end-effector performs some application tasks and during that moves with velocity slowly changing absolute value. Later, it is shown that a cruising motion is time-optimal when at least one of the joint velocity values is at its limit value.

In Figure 2.1 transient and cruising motions are shown together.

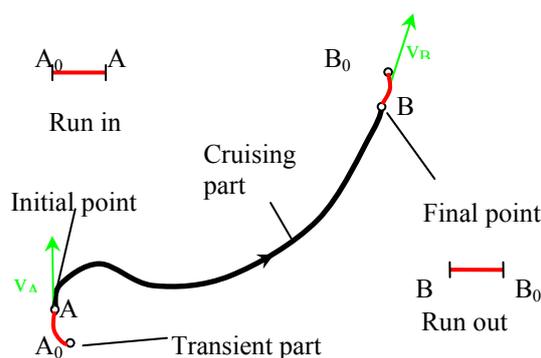


Figure 2.1 The cruising and transient parts of a path.

In Somló J. and Poduraiev J. (1993) [11] a method is presented where, using the parametric method, it is solved that both acceleration and deceleration of the robot motion are at their limit values. in the transient motion parts (see also: Somló J., Lantos B.,P.T. Cat (1997) [2] ).

The proposals for trajectory planning in technical literature are based on rather simple approaches. Below, the method proposed in K.S. Fu, R.C. Gonzalez, C.S.G. Lee (1987) [12] is discussed. (Similar approach is outlined in L. Sciavicco and B. Siciliano [13]).

At this approach, the coordinates of a series of points in Cartesian coordinate system. are given The corresponding joint coordinates values are determined by inverse transformation. If the joint positions, speeds and possibly accelerations (deceleration) are known in the given points (and, also, the desired time of motion from point to point), the paths for joints satisfying the given conditions can be determined using proper-order splines.

For example, if in two points the  $q_i(t_i), \dot{q}_i(t_i), q_i(t_{i+1}), \dot{q}_i(t_{i+1})$  joint coordinates and speed values are given, a third-order spline

$$q_i(t) = a_{0i} + a_{1i}t + a_{2i}t^2 + a_{3i}t^3 \quad (2.1)$$

may be used for path determination of the motion. Because

$$\dot{q}_i(t) = a_{1i} + 2a_{2i}t + 3a_{3i}t^2; \quad (2.2)$$

the  $a_{0i}, a_{1i}, a_{2i}, a_{3i}$  parameter values can be determined from the 4 equations obtained at  $t = t_i$  and  $t = t_{i+1}$ .

When the accelerations are also specified, fifth-order spline with six adjustable parameters can be used. This method may also be used if  $N$  points, termed paths poits, are specified along the paths (see: [13]).

These approaches are rather simple, but need some justification. Sometimes it may turn out that the trapezoidal speed profile is the adequate solution. This can be the case, for example, when the technological process constrains the speed. But, it can turn out only after the motion features were analyzed in detail. Indeed, in general, the speeds, the time of motion between the points, the acceleration, deceleration relations are unknown. In fact (see later) this quantities (among other factors) depend on the configuration of the paths. So, the proper order of investigations is to try to determine, in a systematic way, the above quantities, then go on with the solution of planning problems. The time-optimal cruising trajectory planning method outlined below solves these problems.

### 3 TIME-OPTIMAL CRUISING TRAJECTORY PLANNING

#### 3.1 MOTION ON A GIVEN PATH

First we analyze the case when the path to move on is given. We assume that the acceleration, deceleration abilities of the robot are so high that the transient motion part (see: Figure 2.1.) may be neglected. This condition, usually, is valid for most of the industrial robots and applications. First, in the next paragraph of the present paper we will discuss one of the simplest cases to demonstrate the basic ideas.

##### 3.1.1 Time-optimal cruising motion planning for polar manipulator

In Figure 3.1, a 3-degrees of freedom cylindrical robot is shown. In Figure 3.2 the rotational and horizontal translation degrees are given. We name this mechanism as polar manipulator.

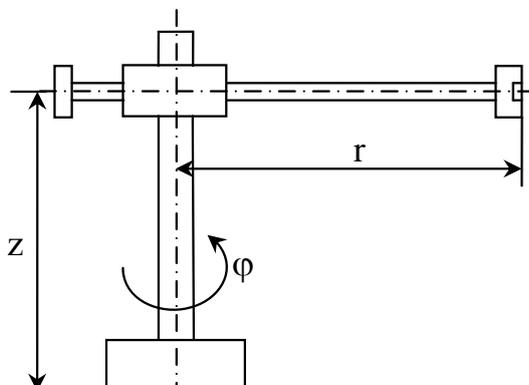


Figure 3.1 Cylindrical robot.

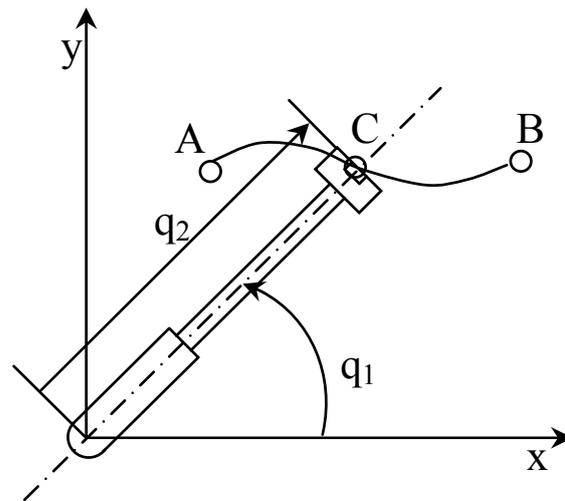


Figure 3.2 Polar manipulator.

For time-optimal cruising trajectory planning we proposed a general method in [2 and 11]. We outline the basic idea of this method below. We want to move the end-effector working point (point C) from point A to point B along the path indicated in the Figure.

The equations of the direct geometry are:

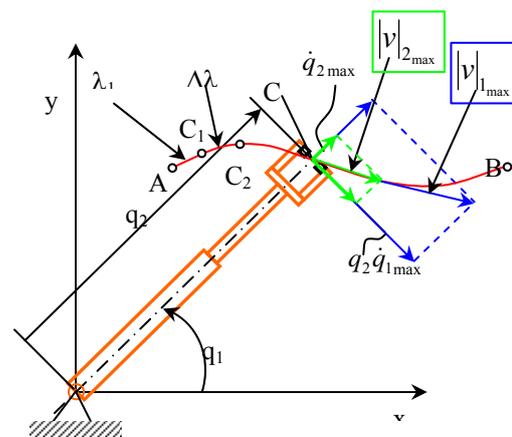


Figure 3.3 Time-optimal motion.

$$\begin{aligned} x &= q_2 \cdot \cos q_1 \\ y &= q_2 \cdot \sin q_1 \end{aligned} \tag{3.1}$$

To realize the motion we need the equations of the inverse geometry. These are:

$$\begin{aligned} q_1 &= \arctan \frac{y}{x} \\ q_2 &= \sqrt{x^2 + y^2} \end{aligned} \quad (3.2)$$

In any given point, the velocity is directed tangentially to the path. The absolute value of the velocity is determined by the components given by the joints. Namely, the rotational joint results a component

$$|v|_1 = q_2 \dot{q}_1 \quad (3.3)$$

The translation joint motion results a component

$$|v|_2 = \dot{q}_2 \quad (3.4)$$

The absolute value of the velocity is

$$|v| = \sqrt{|v|_1^2 + |v|_2^2} \quad (3.5)$$

Now, let us try to determine the possible maximum absolute value of velocity. Let  $\dot{q}_{1\max}$  and  $\dot{q}_{2\max}$  be the maximum value of joint velocities. Clearly, in order to increase the absolute value of the velocity we should increase the joint velocities. Let us consider the case demonstrated in Figure 3.3.

Increasing  $\dot{q}_2$  to his maximum value  $\dot{q}_{2\max}$

$$|v|_{2\max} = \sqrt{(q_2 \dot{q}_1)^2 + (\dot{q}_{2\max})^2} \quad (3.6)$$

The components of the velocity vector are interconnected because the motion should be directed along the tangent to the path. So, to get  $|v|_{2\max}$  is very easy as it is demonstrated on Figure 3.3.

(If analytical form is required, the following relation can be used to determine the required quantities

$$\frac{|v|_1}{|v|_2} = |\tan(\alpha_c - q_1)| \quad (3.7)$$

Here  $\alpha_c$  is the angle of the path related to axis x (see,  $\alpha$  on Figure 3.3). It will be shown bellow that this step is unnecessary to perform.)

Increasing the absolute value of velocity more the maximum of other component may be reached

$$|v|_{1\max} = q_2 \dot{q}_{1\max} \quad (3.8)$$

It is clear that this value may not be realized because at that the other component would exceed its limit value.

So, the optimal velocity value is

$$|v|_{opt} = \min[|v|_{1\max}; |v|_{2\max}] \quad (3.9)$$

In the given case this is  $|v|_{2\max}$ .

Let us now try to get an analytical expression for the realization of the above.

By the derivation of (3.1) we get

$$\begin{aligned} \dot{x} &= -q_2 [\sin(q_1)] \dot{q}_1 + [\cos(q_1)] \dot{q}_2 \\ \dot{y} &= q_2 [\cos(q_1)] \dot{q}_1 + [\sin(q_1)] \dot{q}_2 \end{aligned} \quad (3.10)$$

Solving (3.10) for  $\dot{q}_1$  and  $\dot{q}_2$  we have

$$\begin{aligned} \dot{q}_1 &= \frac{x\dot{x} + y\dot{y}}{\sqrt{x^2 + y^2}} \\ \dot{q}_2 &= \frac{x\dot{y} - y\dot{x}}{x^2 + y^2} \end{aligned} \quad (3.11)$$

Let in a point (for example in C) world coordinates of which are  $x_i$  and  $y_i$  the angle of the tangent of the path with axis x be  $\alpha_i$  (earlier we used for the same  $\alpha_c$ ). Then

$$\dot{x} = |v| \cos \alpha_i \quad \text{and} \quad \dot{y} = |v| \sin \alpha_i \quad (3.12)$$

Substituting the quantities into (3.11) we get

$$\begin{aligned} \dot{q}_1 &= S_1(x_i, y_i, \alpha_i) |v| \\ \dot{q}_2 &= S_2(x_i, y_i, \alpha_i) |v| \end{aligned} \quad (3.13)$$

where  $S_1$  and  $S_2$  are the quantities obtained by the substitutions. So, for the maximum values we get

$$\begin{aligned} \dot{q}_{1\max} &= S_1(\dots) |v|_{1\max} \\ \dot{q}_{2\max} &= S_2(\dots) |v|_{2\max} \end{aligned} \quad (3.14)$$

Accordingly,

$$\begin{aligned} |v|_{1\max} &= \frac{\dot{q}_{1\max}}{S_1(\dots)} \\ |v|_{2\max} &= \frac{\dot{q}_{2\max}}{S_2(\dots)} \end{aligned} \quad (3.15)$$

For the determination of the optimal velocity value Equation (3.9) is valid.

According to the above, in every point of the paths the time-optimal cruising velocity can be determined. This

velocity depends on the geometry of the paths (world coordinates of the points and direction of the tangents to the paths) and on the maximum possible values of the joint velocities.

We name *dominant* the joint the maximum velocity of which determines the optimum. The dominance may change in some points. In these points both joints result the same maximum velocity (for the given example in these points  $|v|_{1\max}$  and  $|v|_{2\max}$  are equal).

In fact, Equations (3.10) and (3.11) are the rows of the Jacobian matrices. So, all the above can be interpreted from the point of view of using Jacobian matrices. This has been done in Somló, Lantos, P.T.Cat [2]. In this case we use

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{J}(\mathbf{x})\dot{\mathbf{q}} \\ \mathbf{x} &= (x, y)^T \\ \dot{\mathbf{q}} &= \mathbf{J}^{-1}(\mathbf{q})\dot{\mathbf{x}} \\ \mathbf{q} &= (q_1, q_2)^T\end{aligned}\quad (3.16)$$

where  $\mathbf{J}$  and  $\mathbf{J}^{-1}$  are the Jacobian and inverse Jacobian matrices respectively.

### 3.1.2 Parametric method of trajectory planning

It was mentioned above that we propose to solve the time-optimal cruising trajectory planning for the whole path but how to do that was not outlined in a systematic way. In what follows, we will try to solve this task.

Shin and McKay in [7 and 8] proposed to use a parametric approach for robot planning problem. They proposed as a parameter the path length. It is clear that at any form of description of any path the world coordinates may easily be expressed as functions of path lengths. Indeed, the distance of any two points in plane or in 3D space may be easily computed. Representing the world coordinates of a robot as function of the determined paths lengths, the parametric description problem is solved. The inverse transformations connect the joint coordinates values with the world coordinates values. So, if the last ones are expressed by the parameter, it leads straight to the opportunity to express also the joint coordinates as functions of the parameter. This is the essence of the parametric approach. In fact, the parameter is an independent variable for the planning.

Let us return to the analyzed above example. Let the world coordinates in a point of the path be  $x_i$  and  $y_i$ . Let in the next (nearest) point of the path the coordinates be  $x_{i+1}$  and  $y_{i+1}$  .. The distance between the two points is

$$\Delta\lambda_{i+1} = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \quad (3.17)$$

and

$$\alpha_i = \arctan \frac{y_{i+1} - y_i}{x_{i+1} - x_i} \quad (3.18)$$

For the point with index (i-1) we have

$$\Delta\lambda_i = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2} \quad (3.19)$$

Let us assign to every point which we consider a serial number. This numbers are:  $i = 1, 2, \dots, N$ , and are also used as indices of the points. The last index value is  $N$ . The length of the path until point with index  $k$  is indicated as  $\lambda_k$ . Clearly  $\lambda_1 = 0$ . For others

$$\lambda_k = \sum_{i=2}^k \Delta\lambda_i \quad (3.20)$$

In the last point of the path  $k = N$  and  $\lambda_N$  is the overall length of the path for which we will simply use  $\lambda$ .

In any point of the path the  $\lambda_k$  value can be computed. To  $\lambda_k$  belong the  $x_k$  and  $y_k$  world coordinates values. By inverse transformation the  $q_{1k}$  and  $q_{2k}$  values may be computed. In such a way the following functions may be determined:

$$\begin{aligned}q_1 &= f_1(\lambda) \\ q_2 &= f_2(\lambda)\end{aligned}\quad (3.21)$$

Of course, in this way we obtain functions determined in discrete points. This is very much suitable for robot control tasks where, usually, similar representations are used. But, if necessary, sometimes, analytical relations may also be developed. Let us consider the above example, and motion along a straight line (from A to B; Figure 3.3).

In this case

$$\begin{aligned}x &= x_A + \lambda \cdot \cos(\alpha) \quad \text{and} \quad y = y_A + \lambda \cdot \sin(\alpha) \\ \arctan \frac{y_A + \lambda \cos \alpha}{x_A + \lambda \sin \alpha} \\ \text{and } q_2 &= f_2(\lambda) =\end{aligned}\quad (3.22)$$

$$\sqrt{(x_A + \lambda \cos \alpha)^2 + (y_A + \lambda \sin \alpha)^2}$$

Assuming  $\lambda = k \cdot \Delta\lambda$  ( $k = 1, 2, \dots, N$ ), we return to the discrete representation.

We remark that in this way (if necessary) sub-division of interpolation sections of paths is possible.

Now, we analyze the discrete velocity values in the selected points of the path.

$$\begin{aligned}\frac{\Delta q_1}{\Delta t_k} &= \frac{\Delta f_1}{\Delta t_k} = \frac{f_1(\lambda_k + \Delta\lambda_k) - f_1(\lambda_k)}{\Delta\lambda_k} \frac{\Delta\lambda_k}{\Delta t_k} \\ \text{and} \\ \frac{\Delta q_2}{\Delta t_k} &= \frac{\Delta f_2}{\Delta t_k} = \frac{f_2(\lambda_k + \Delta\lambda_k) - f_2(\lambda_k)}{\Delta\lambda_k} \frac{\Delta\lambda_k}{\Delta t_k}\end{aligned}\quad (3.23)$$

It can be recognized that the velocity values may not exceed their limit values. So,

$$\dot{q}_{j\max} = \left( \frac{\Delta f_j}{\Delta t} \right)_{\max} \quad (j = 1,2) \quad (3.24)$$

$$\frac{\Delta f_j}{\Delta \lambda} = \frac{f_j(\lambda + \Delta \lambda) - f_j(\lambda)}{\Delta \lambda} \quad (j = 1,2) \quad (3.25)$$

$$|v|_{j\max} = \left( \frac{\Delta \lambda}{\Delta t} \right)_{\max} \quad (j = 1,2) \quad (3.26)$$

Based on all above

$$|v|_{j\max} = \frac{\dot{q}_{j\max}}{\frac{\Delta f_j}{\Delta \lambda}} \quad (j = 1,2) \quad (3.27)$$

Then,

$$|v|_{opt} = \min \{ |v|_{j\max} \} \quad (j = 1,2) \quad (3.28)$$

It is easy to recognize that if for a 3D problem we use the parametric representation, then we get

$$q_i = f_i(\lambda) \quad (i = 1,2,3) \quad (3.29)$$

The derivatives of the joint coordinates may be obtained as

$$\frac{dq_i}{dt} = \frac{\partial f_i}{\partial \lambda} \frac{d\lambda}{dt} \quad (i = 1,2,3) \quad (3.30)$$

$$\left( \frac{d\lambda}{dt} \right)_{i\max} = |v|_{i\max} = \frac{\dot{q}_{i\max}}{\frac{\partial f_i}{\partial \lambda}} \quad (i = 1,2,3) \quad (3.31)$$

The  $\frac{\partial f_j}{\partial \lambda}$  values may be computed as indicated above (see

Equation (3.25)).

The time-optimal velocities are

$$|v|_{opt} = \min \{ |v|_{i\max} \} \quad (i = 1,2,3) \quad (3.32)$$

For the mathematical correctness of what mentioned above, it should be emphasized that the relations are meaningful in the points only where the derivatives used exist. So, the cross points of interpolation sections, the singularity points should be excluded from considerations. The above fact does not give any difficulty from the point of view of practical realizations.

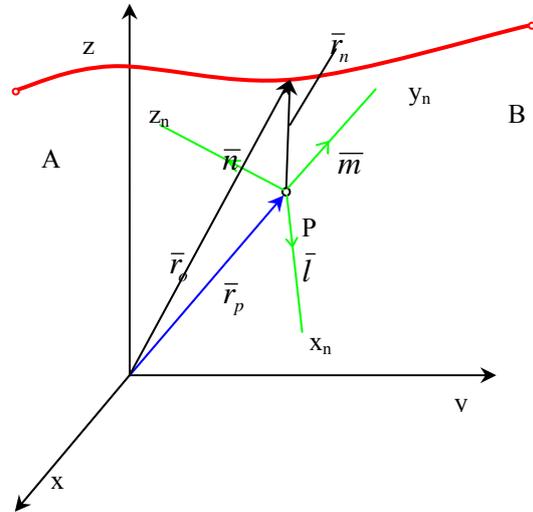


Figure 3.4 Changes of positions and orientations.

Returning to the above, by determining the  $|v| = \dot{\lambda}(\lambda)$  function for the whole path, in fact, we solve the trajectory planning problem. A graphic of a curve obtained for some planning tasks is demonstrated in Figure 3.5. This curve in itself is very much meaningful. At any  $\lambda$  value the velocity along the path may not be higher than the corresponding  $\dot{\lambda}$  on the curve. On the contrary, any value below the curve may be applied. If a constant velocity along the path, is required for the whole path it may not be higher than  $\dot{\lambda}_{\min}$ . In some cases, different constant velocities along the path sections may be required. These may be constructed using the curve. Other opportunities are possible, too.

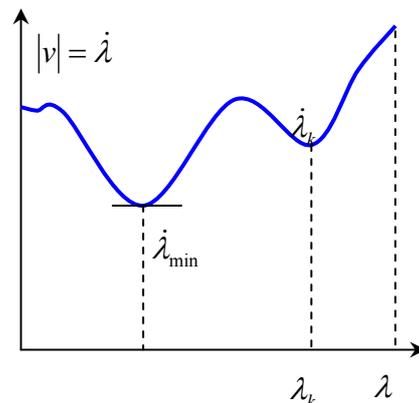


Figure 3.5 A  $|v| = \dot{\lambda}(\lambda)$  function.

But the most important fact is that using the planning result we can establish the motion time path length relation.

Indeed,

$$\frac{d\lambda}{dt} = |v(\lambda)| \quad (3.33)$$

So

$$t(\lambda) = \int_0^\lambda \frac{d\lambda}{|v(\lambda)|} \quad (3.34)$$

By the use of the Equation (3.34) the time values belonging to any  $\lambda$  can be determined and so the

$$q_1 = g_1(t) \text{ and the } q_2 = g_2(t) \quad (3.35)$$

input signals for the drives may be determined which realize time-optimal cruising motion.

An example for the polar manipulator is given in the Appendix.

### 3.1.3 The changes of position and orientation together

In the previous paragraph only the translation motion of the end-effector centre point was considered. It is well known, that together with the translation motions, depending on the path, on the robot construction and parameters, the orientation also changes. Requirements are formulated, very frequently, not only for translation but for orientation motion, too. The proposed method can be used for this case without any change.

The synchronised translation and orientation changes can be interpreted as shown in Fig. 3.6.

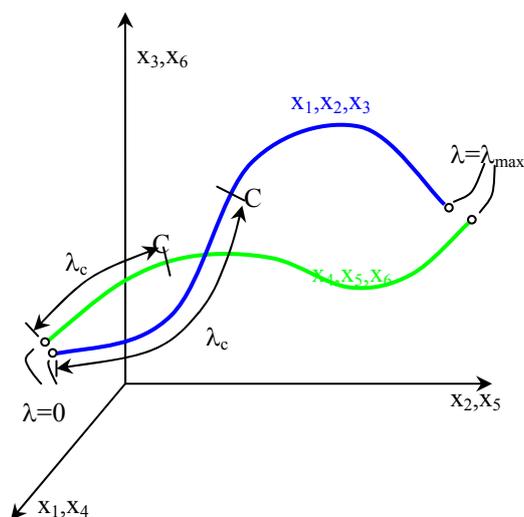


Figure 3.6 Translation and orientation paths.

The tool-centre point motion is characterized by vector  $\mathbf{r}_p$ . The tool-frame  $x_n, y_n, z_n$  is determined by its orthogonal unit vectors  $\mathbf{l}, \mathbf{m}, \mathbf{n}$ . As it is well known there are only 3

independent values among the components of these unit vectors. So, if

$$\begin{aligned} \mathbf{l} &= (l_x, l_y, l_z)^T \\ \mathbf{m} &= (m_x, m_y, m_z)^T \\ \mathbf{n} &= (n_x, n_y, n_z)^T \end{aligned} \quad (3.36)$$

Then, selecting any 3 independent components we can introduce the  $x_4, x_5, x_6$  orientation (world) coordinates. To be consequent, we introduce also  $x = x_1, y = x_2, z = x_3$ . Solving the path planning problem, which now can be interpreted in spaces  $x_1, x_2, x_3$  and  $x_4, x_5, x_6$ , we can solve the trajectory planning problem. For that we use the parametric method.

The parametric method of planning gives:

$$q_4 = f_4(\lambda), q_5 = f_5(\lambda), \text{ and } q_6 = f_6(\lambda) \quad (3.37)$$

We remark that, usually, in the equations of the inverse geometry, for  $q_4, q_5, q_6$  the values of  $q_1, q_2, q_3$  are involved. It does not give any difficulty when using the parametric method.

All the above is a clear indication for the fact that the time-optimal trajectory planning method and equations stay valid. That is Equations (3.31) and (3.32) results time-optimal motion with the only change

$$i = 1, 2, 3, 4, 5, 6 \quad (3.38)$$

We remark that when 5D application problems are solved by the robots the orientation space becomes a plane. When the orientation should be kept constant the orientation curve becomes a point. Irrespective of the above, the proposed planning method stays valid.

## 4 TIME-OPTIMAL MOTION FROM POINT TO POINT

The point-to-point (PTP) motion is simpler to analyze and realize than that for motion along a path. Nevertheless, it is very important to understand and use time-optimal motions. When PTP motion is used, proper step functions are input to the actuators of the joints, and the realized motions of the joints are step responses. The motions in Cartesian coordinates systems are determined by the equations of direct geometry of the robot. If the transient processes are short and the joints motions realize maximum velocities, the motion is close to same time-optimal cruising one but with undetermined before geometry. Now, let us analyze the question what are the opportunities of realizing the point-to-point motion when the motion on different paths may be realized, too.

As an example, consider again the rotating and translating joint of a cylindrical robot (Fig. 3.2). Let us suppose that the transient motion is very fast. So, the motion features are determined basically by the cruising.

Let the task be to get from point A to B of Fig. 3.2. Then the difference of coordinates is:

$$\Delta q_1 = q_{1B} - q_{1A} \text{ and } \Delta q_2 = q_{2B} - q_{2A} \quad (4.1)$$

Let the velocity limits be

$$\dot{q}_{1\max} \text{ and } \dot{q}_{2\max} \quad (4.2)$$

They are constant values, independent of the states. Let

$$t_{1\min} = \frac{\Delta q_1}{\dot{q}_{1\max}} \text{ and } t_{2\min} = \frac{\Delta q_2}{\dot{q}_{2\max}} \quad (4.3)$$

It is clear that only the bigger of the two values in (4.3) may be realized. This will give the absolute minimum of the motion time.

In formal terms

$$t_{\min} = \max(t_{i\min}) \quad i = 1,2 \quad (4.4)$$

The joint which determines this minimum time is named *dominant*.

At minimum time motion the dominant joint will be at its maximum possible velocity in every moment.

The other joint's velocity value may vary in the domain

$$\dot{q}_{j\min} \leq \dot{q}_j \leq \dot{q}_{j\max} \quad j = 1,2 \quad (4.5)$$

Now, let the dominant joint index be  $i$  and the non-dominant joint index  $j$ . Let us move the dominant joint with the velocity  $\dot{q}_{i\max}$ . It is clear that any velocity for the other joint in the  $\dot{q}_{j\min}, \dot{q}_{j\max}$  domain (see (4.5)) may be applied.

If moving from point A with  $\dot{q}_{j\min}$  and  $\dot{q}_{j\max}$  the borders of minimum time paths in the  $x, y$  plane may be determined. Using the equations of kinematics it is very easy to get the boundary curves. It is equally easy to get the boundaries for the motion from point B using the method of backward time. (see Fig. 4.5). If there exists a common, contiguous area inside the borders containing A and B, then every realizable trajectory of this area is a minimum time one.

Any trajectory of the domain is realizable if  $q_i = \dot{q}_{i\max}$  and the (4.5) constraint ( $j = 2$ ) is satisfied.

Let us determine the border curves for the above example:

a) Case when the rotation motion is dominant.

That is:  $t_{1\min} > t_{2\min}$ .

The speed of rotating joint should always be at its limit value.

$$\dot{q}_1 = \dot{q}_{1\max}$$

For the motion of the translating joint one has the following limit values:

For the motion from point A

$$q_{2A} - \frac{q_1 - q_{1A}}{\dot{q}_{1\max}} |q_{2\min}| \leq q_2 \leq q_{2A} + \frac{q_1 - q_{1A}}{\dot{q}_{1\max}} \dot{q}_{2\max} \quad (4.6)$$

For the motion from (to) point B

$$q_{2B} - \frac{q_{1B} - q_1}{\dot{q}_{1\max}} \dot{q}_{2\max} \leq q_2 \leq q_{2B} + \frac{q_{1B} - q_1}{\dot{q}_{1\max}} |q_{2\min}| \quad (4.7)$$

b) Case when the translation motion is dominant.

That is:  $t_{2\min} > t_{1\min}$ .

The velocity of the translation motion should always be at its limit value.

$$\dot{q}_2 = \dot{q}_{2\max}$$

For the motion of the rotating joint one has the following limit values:

For the motion from point A:

$$q_{1A} - \frac{q_2 - q_{2A}}{\dot{q}_{2\max}} |q_{1\min}| \leq q_1 \leq q_{1A} + \frac{q_2 - q_{2A}}{\dot{q}_{2\max}} \dot{q}_{1\max} \quad (4.8)$$

For the motion from (to) point B:

$$q_{1B} - \frac{q_{2B} - q_2}{\dot{q}_{2\max}} \dot{q}_{1\max} \leq q_1 \leq q_{1B} + \frac{q_{2B} - q_2}{\dot{q}_{2\max}} |q_{1\min}| \quad (4.9)$$

The realizable trajectories lie inside the border curves demonstrated on Fig. 4.1 and 4.2. All the trajectories for which in every point the following relation is fulfilled:

$$\dot{q}_{i\min} \leq \dot{q}_i \leq \dot{q}_{i\max} \quad i = 1,2, \quad (4.10)$$

and have the given initial and final points are realizable trajectories.

The border curves are also realizable trajectories. There is always a special realizable trajectory which results the termination of the motions for the different joints at the same time. In the case of the given example, it simply means that the quicker joint velocity should slow down to have  $t_2 = t_{1\min}$  (or  $t_1 = t_{2\min}$ ). This implies the corresponding trajectory inside the border curves.

It is an interesting question how to choose from the set of realizable trajectories. All these trajectories realize the minimum of the time of the motion. It is possible to find trajectories which have some features more favorable than the others. For example: it can be a criterion to find the minimum time trajectory realizable by the minimum of energy.

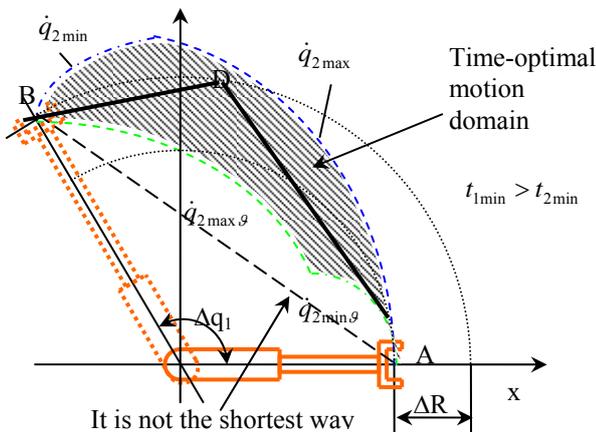


Figure 4.1 Time-optimal PTP motion.

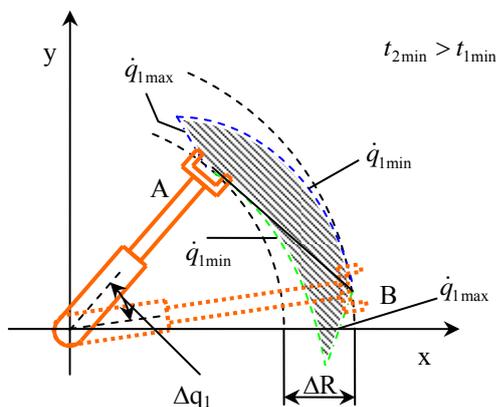


Figure 4.2 Time-optimal PTP motion.

It is also an interesting question what is the geometry of the minimum time trajectories for different robots and different tasks. It can be seen in Fig. 4.6 that, for example, in the case of Fig. 4.6 b there is a straight line path which can be a minimum time, realizable trajectory. In the case of Fig. 4.6 a, such a line does not exist. The behavior of the robots from this point of view, including the investigation of the time of motions on different arcs, was analyzed by H. Doghiem (1993 a, 1993 b) [14,15].

#### 4.1 Case when the shortest way for a robot between two points is not the straight line

Looking at Fig.4.1 it can be recognized that if moving from A to B along the straight line it will take more time than moving on any time-optimal cruising trajectory. So, if we regard as the shortest way the one which takes the least time, it will not be the straight line. To handle the case is very easy. First we define the minimum time using Relation (4.4). Then we determine the time-optimal cruising trajectory planning for the straight line, which gives also the motion time, we get a clear picture about the quantitative characteristics.

#### 4.2 Optimal PTP motion in space

What outlined for the example can be easily generalized for any degree of freedom. The dominant joint determination is exactly the same as in 2D case. Of course, when ND ( $N=3, 4, 5, 6$ ) problems are considered in Equations (4.1) ÷ (4.5), the quantities with the proper indexation should be used. Furthermore, the individual representation of quantities for translation motion of the tool-centre point and orientation motions is necessary (as it was introduced for the investigation of time-optimal motions on a given path). Assuming that the dominant joint moves with the limit speed, the other joints have some freedom of motion the limits of which can be determined similarly as in the 2D example above. So, we move the dominant joint by the maximum velocity. Moving the other joints with  $\pm$  limit velocities (from point A and with backward time from point B) we get sub-spaces which form time optimal domains. Any realizable trajectory of these sub-spaces is a time optimal trajectory. The optimal motion paths domains now become 3D sub-spaces for translation and orientation motions. It is not so easy as in the 2D case to analyze the processes but it is fully possible. Even, graphical representations may help to find the most suitable paths.

#### 4.3 Optimal motions in free space with obstacles

An excellent opportunity to solve the motion planning problem in the presence of obstacles is given by the approach outlined above. In this case, it is highly proposed to use graphical representations of the time-optimal motion domains. If realizable paths exist in between the optimal border and the obstacles borders, all these are time-optimal, that is give the same minimum time. The final choice may be made by considering a number of other than the time aspects.

### 5 REALIZATION OF TIME-OPTIMAL MOTIONS

Let us deal at first with the motions in a given paths. As it is clear from the outlined the determination of the time-optimal trajectories is extremely simple. The equations which are necessary, for most of the industrial robots are known or can be easily derived. The maximums of joint velocities are included in the user handbooks of the robots. If in doubt, these values can easily be measured.

The optimal trajectory planning computations may be easily realized off-line (see: Appendix). Then, using the data, final choice of the velocities is possible with the guarantee of part processing times. These may prove much more favorable than the ones chosen without the knowledge of consequences of the planning decisions. For the realization of what above detailed user oriented, interactive computer programs can and were developed.

For the demonstration, let us consider the Figure in Appendix. It can be recognized that moving from point A until close to point C the velocity value along the path is close to  $|v|=0.1 \text{ m/s}$ . In the region close to point D this velocity may reach a triple value. Many similar facts can be

recognized at the solution of individual tasks. Conclusions may be deduced for application planning, concerning even the path planning issues.

Returning to the possibility of realizing time-optimal (or close to that) motions, it can be stated that the modern industrial robot motion characteristics (velocities) are so high that even the minimum of time-optimal velocities is so high for the given paths that this constant velocity may be applied very effectively. But it is useful to know what it is. One may never know when the big problems occur. So, if a robot control is equipped with an option which realizes time-optimal motion, it is a significant step forward. To demonstrate the opportunity of realizing such a control system, an open system architecture robot control device was developed and experimented by Sokolov, Somlo, Lukanyin as reported in publications [16 and 17].

It is remarkable that the time-optimal motion planning gives upper bounds for the velocities. The final choice of the velocity should take into account other constraints, too (see: Somlo, Lantos, P.T.Cat [2]). The other constraints may be grouped into: technological constraints, safety constraints, psychological constraints, etc.

Let us return to the question of free paths. For simplicity let us consider the 2D case, but suppose, that the device is able to realize not only PTP motions. Let us try to determine the path of the time-optimal motion. First (trivial) choice may be the straight line between A and B. If it is not time-optimal (see: Figure 4.1), we can try to choose two straight line sections (ADB in Figure 4.1). If these are realizable, they might construct the path. Application of third and higher order splines for motion planning was reviewed in paragraph 2.1. Splines may also be candidates for realizing time-optimal trajectories. For satisfying the time-optimality condition they should fully lie inside the optimal border, and the trajectories should be realizable.

## 6 PATH, TRAJECTORY PLANNING AND TRAJECTORY TRACKING TOGETHER

Path and trajectory planning tasks are strongly interconnected. If the planner allocates in the workspace of the robot some paths, it is a very easy task to determine the time-optimal cruising regimes. It is also possible to determine the corresponding  $\dot{\lambda}(\lambda)$  functions. Estimating these functions gives a through picture about velocity characteristics. Velocities may be properly decreased, manipulated. Paths may be relocated in the workspace. Paths may be modified (for example, smoothed to get smaller velocity differences), etc.

Trajectory tracking should provide the realization of the inputs given by trajectory planning. We divide the problem into two parts. The first part is the transient part realization, while the second comprises the realization of the motion on the time-optimal cruising part. As it was mentioned, to analyze the motion on the transient part and find suitable (optimal) control laws are not easy tasks and hard to realize. Nevertheless, this field is full of nice results.

Below, we will deal only with the cruising part. As, for every point of the paths we have the joint coordinates, and correspondingly, all the derivatives of them, we can determine, (using the Lagrange equation (1.1), (1.2) (or others)) the torques (forces) necessary for the given motion. If all the torques (forces) are in the available regions (below the limit values) there is a good chance that the robot control will provide processes close to the required. This is especially true if sophisticated robot control methods are used (see: Asada, Slotine [1], Somlo, Lantos, P.T. Cat [2], Fu, Gonzalez, Lee [12], Sciavicco, Siciliano [13] and many others).

Concerning trajectory tracking, the control science provides effective methods for the solutions. For example: Computed Torque, Model Reference Adaptive, Sliding Mode Control may be used (see, for example: [1,2,4,5,12,13]).

## 7 ONE FIELD OF POSSIBLE APPLICATIONS

To use low depth very high feed motions is a great promise for many technological processes. We are concerned with the problem of making corsets for spinal disorders (see: Somló, Tamás, Halász [20] and others, for example, [18] and [19]) The aim of the works is to develop an "intelligent corset" (plastic girdle) and a medical examination appropriate for early recognition and treatment of spinal disorders of young people aged 8-18 including a multidisciplinary methodology on the basis of medical science, informatics and material technologies. These research tasks also include the investigations of the opportunities of using advanced CAD/CAM technologies for the realization of the corsets. Corsets are made from plastic sheets that can not be produced by classical CNC machine tools technologies. Usually, dies are produced which serve for making the parts by pressing or by vacuum forming, etc. Recently, a new technology called. DSF (Dieless Sheet Forming; sometimes ISF (Incremental Sheet Forming) is used). has appeared to manufacture individual free-form sheets (In the literature Sheetmetal forming is in use but we intensively left out the metal attribute.) The principle was published in the patent of E. Leszak in 1967 [21]. EU Sixth Framework Project (Contract:014026) was devoted to the topic [22]. A survey of Japans results is published in [23].

In Figure 7.1 DSF process, equipments and product is demonstrated.

In Figure 7.2 one-sided forming with warming is shown.

The DSF technology is very similar to CNC manufacturing processes. It is similar, most typically, to vertical milling of sculptured surfaces by finger-like milling tool. However, no cutting but pressing or hitting of the surface is performed.

In our research work polyethylene sheets were formed. The reason was, as it was noted above, that such sheets might be useful at corset making.

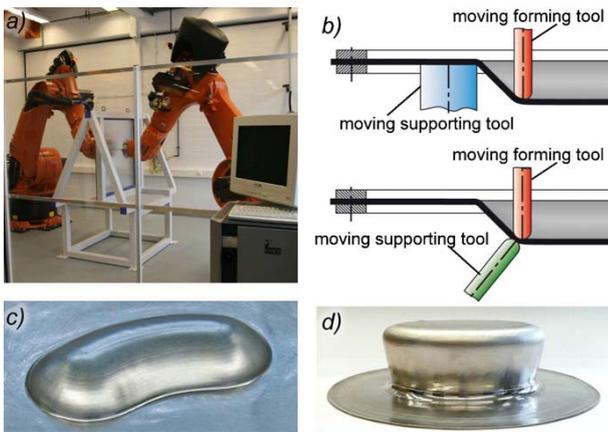


Figure 7.1 DSF process and equipment.

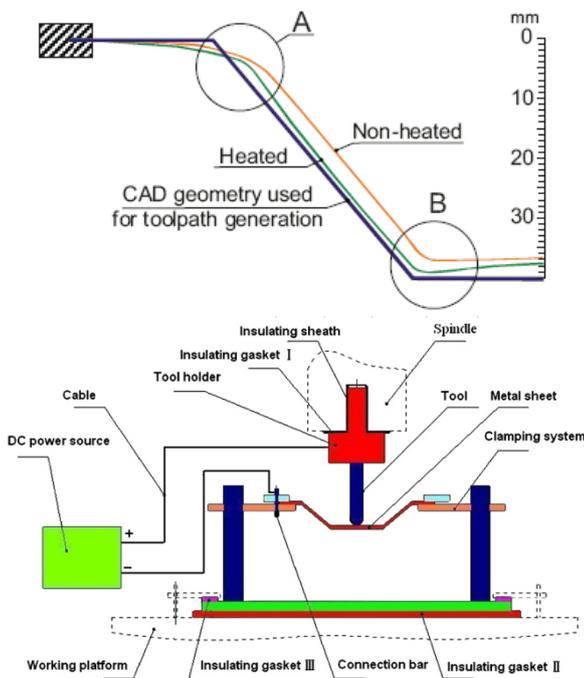


Figure 7.2 One-sided forming with warming.

In Figure 7.3 a corset and a girl with spinal disorder are shown.



Figure 7.3 A corset and a girl with spinal disorder.

Prior to our experiments, we could get information on DSF of polymer materials in cool condition only. For us it was clear, mostly from the practice of corsets making, that in our case warming was necessary. It was a long way to find the proper temperatures and processing data. After finding the proper regimes, we could realize simple parts (truncated cones) in high quality. Then, we began to produce trough-like polyethylene parts. The thickness of sheets was  $3 \div 5$  mm. The experience gained by the cones was very useful. In a very short time we could find the proper warming regime and data to get good-quality parts. In this case we faced other problems. The method of fixation of the part affected the quality very much. The sheet should have fixation stripe close to the forming actions. This is an indication for the fact that if no two-sided processing is used, some additional measures are necessary to provide for satisfying the requirements. After solving all these problems the trough-like part making provided suitable quality. But the time of production was not short enough. For wide practical use the production time should be significantly decreased.

It seems to us that the given technological process is very much suitable for trying to use very high, possibly time-optimal velocities.

One of the reasons for that is that the contact forces are extremely low.

Similar conditions are valid when sculptured parts are produced from polyfoam materials by cutting. This is also used at corset making. Polyfoam torsos are manufactured by CNC machine tools which serve for vacuum forming of corsets. These torsos may also be produced by high velocity cutting realized by robots.

## 8 CONCLUSIONS

Proposals in the paper give a general and simple to-realize method for time-optimal cruising trajectory planning for industrial robots. The proposed approach is based on the parametric method. All the parameters which are needed for the application (for example, joint velocities limit values) are easily available. The basic relations reflecting the essence of the approach are given by Equations (3.29) ÷ (3.32). Then, determining the  $\dot{\lambda}(\lambda)$  function and from that the  $t = t(\lambda)$  relations, the joint drives inputs may readily be determined and consequently the time-optimal motion may be realized. A slightly different but in the spirit close method can be used for free paths.

The time-optimal trajectory planning method provides plenty of information for application planning. Existing applications' time needs may be shortened, and new applications may be developed with outstanding characteristics.

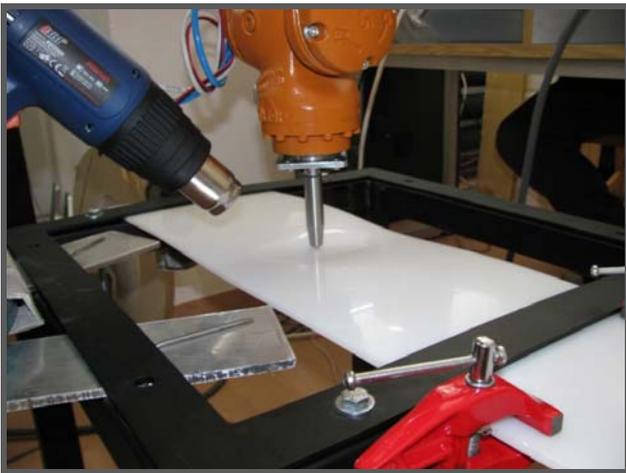


Figure 7.4 DSF of polyetilen sheets by warming.



Figure 7.5 Through-like part made by DSF.

### 8.1 SOME RESEARCH TOPICS

It is reasonable to reconsider existing robot applications and estimate the opportunity of decreasing times.

It is reasonable to relocate the paths in workplace (for existing robot applications) to find out their effect on production time (where to put?).

It is an interesting research problem how close the optimal processes may be approximated by robots having rather simple control opportunities. For example, if piece-wise constant velocities are used for shorter or longer sections (how to program?).

Many other similar problems may be formulated all giving practical advance with almost no price at all.

### ACKNOWLEDGEMENT

Many thanks to Péter Tamás for the help in research work and preparation of the paper.

The authors are grateful to NKTH (Research and Technology Office of Hungarian Government) for the support of the research on spinal disorder corrections (Project: TECH 08-A1/2-2008-0121 (NKTH)).

### APPENDIX

Let us consider a polar manipulator which is given in Figure A1. The shape of the path to be realized is also given (the proportions are not real). The circle centre point data are not indicated because the circle is tangent in point B to line AB.

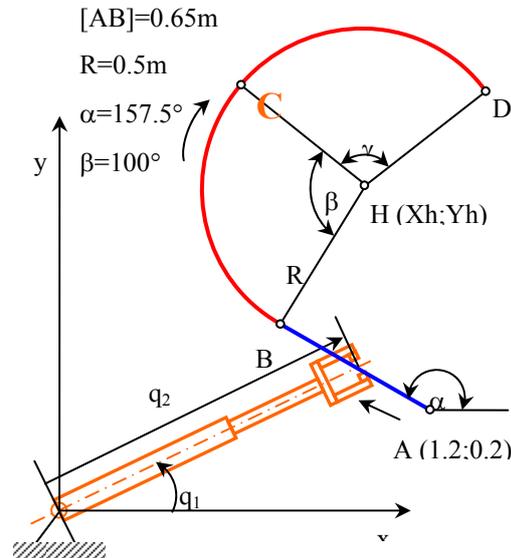


Figure A1 Polar manipulator and given path.

We determined the time-optimal cruising trajectory according to the proposed in the paper. The following observations can be concluded. The motion begins at point A with the maximum (negative) speed for the translation joint. Close to point B (still on the linear part of the path) the maximum speed "switches" to the rotation joint. Then,

the optimal motion velocity is determined by  $\dot{q}_{1max}$ . The length of this section is about 3 sec. The dominance after about 7 sec of the overall motion switches again to the translation joint and a long motion section with  $\dot{q}_{2max}$  velocity is realized, until, close to point D, again the rotation joint becomes dominant.

It is easy to recognize that if we use a constant velocity along the whole path its value should be below  $v = 0.1$  m/s. Close to point D much higher velocities may be used. The overall time of the motion along the path at time-optimal motion is about 17 sec. If  $v = 0.1$  m/s is applied, this time is 27,4 s. So, the economy is significant.

It is noteworthy that it was not mentioned here that the velocity limits of the joints should have constant values. For example, a real constraint might be given by the restriction of the centrifugal force acting on the load. These can be derived from the requirements of the safe gripping.

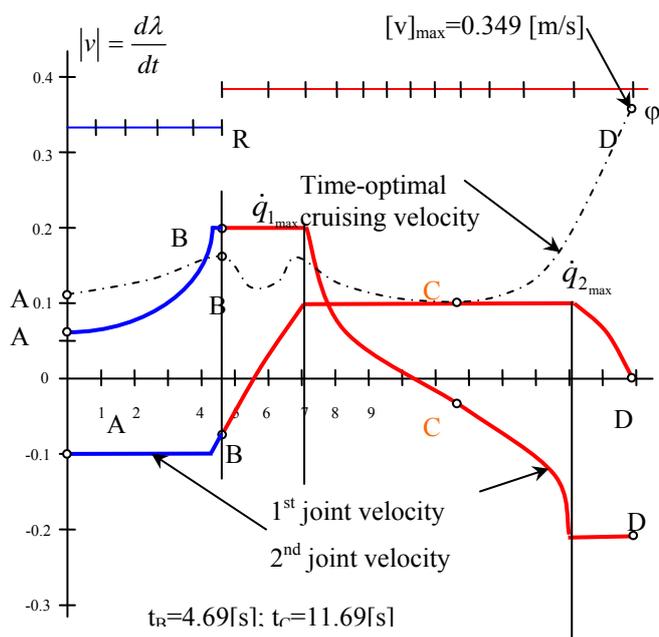


Figure A2 Time-optimal cruising trajectory.

Let

$$q_2(\dot{q}_1)^2 < \frac{GF}{m_p} \quad (A.1)$$

where GF- is the necessary gripping force,  
 $m_p$  - loading mass.

At certain points of the path, (A.1) may give more severe velocity limit than the manual given constraint.

## REFERENCES

- [1] Asada H. and Slotine J.J.E., Robot analysis and control. Wiley & Sons, pp. 266, 1986.
- [2] Somló J., Lantos B. and Cat P.T., Advanced robot control. Akadémiai Kiadó, Budapest, pp. 425, 1997.
- [3] Geering H.P., Guzzella L., Hepner S.A.P. and Onder C.H., Time-optimal motions of robots in assembly tasks. *IEEE Transe. on Automatic Control*, AC-31, N6, 1986.
- [4] Korondi P., Hashimoto H., Gajdar T. and Suto Z., Optimal sliding mode design for motion control. *IEEE International Symposium on Industrial Electronics (ISIE'96)*, pp. 277-282, 1996.
- [5] Korondi P., Young D. and Hashimoto H., Sliding mode based disturbance compensation for motion control. *23rd International Conference on Industrial Electronics, Control and Instrumentation (IECON '97)*, New Orleans, 1997.11.09-1997.11.14, pp. 73-78.
- [6] Singh S.K., Trajectory planning for robot control: a control systems perspectives. *Control Dynamic Systems*, pp. 105-146, 1991.
- [7] Shin K.G. and McKay N.D., Selection of near-minimum time geometric paths for robotics manipulators. *IEEE Trans. of Autom. Contr.*, v. AC-31, pp. 501-511, 1986.
- [8] Shin K.G. and McKay N.D., Minimum-time control of robotics manipulators with geometric path constraints. *IEEE Trans. of Academic Press*, 1985.
- [9] Shin K.G. and McKay N.D., Minimum cost trajectory planning for industrial robots. *Control Dynamic Systems*, Academic Press, pp. 345-403, 1991.
- [10] Podurajev J. and Somlo J., A new approach to the contour following. Problem in robot control (Dynamic Model in Riemann Space). *Mechatronics (GB) V.3., N2.*, pp. 241-263, 1993.
- [11] Somlo J., Poduraev J., Optimal cruising trajectory planning for robots. *IASTED Conference Robotics and Manufacturing*, Oxford, 1993.
- [12] Fu K.S., Gonzalez R.C. and Lee C.S.G., *Robotics: control, sensing, vision, and intelligence*. McGraw-Hill, 1987.
- [13] Sciavicco L. and Siciliano B., Modelling and control of robot manipulators. Springer, pp. 378, 2001.
- [14] Doghiem H., Maximum cruising speed (minimum time) robot trajectory planning. *IASTED Conference Robotics and Manufacturing*, Oxford, Sept, 1993.
- [15] Doghiem H., Minimum time cruising trajectory planning for robots using a dynamic programming approach. *CAMP'93*, Budapest, 1993.
- [16] Sokolov A., Optimal computer control of redundant robots. PhD Dissertation, Budapest University of Technology, 2000.
- [17] Somló J., Sokolov A., Lukanyin V., Automatic trajectory planning for robots. *ES2000*, Portorozs Slovenija. September 17-19, 2000, pp. 129-134.
- [18] Tamas P. and Halasz M., Relationship between basic body sizes and body shape, *4th International Textile, Clothing & Design Conference - Magic World of Textiles*, Dubrovnik, Croatia, October 5<sup>th</sup>-8<sup>th</sup>, 2008.
- [19] Wenzel Klára, Antal Ákos, Molnár József, Tóth Bertalan, Tamás Péter, New Optical Equipment in 3D Surface Measuring. *Journal of Automation Mobile Robotics & Intelligent Systems*, 3:(4), pp. 29-32, 2009.
- [20] Somló J., Tamás P., Halász M., A new application of robotic devices for correction of spinal disorders. *TMT 2009*, Hammamet, Tunisia, October 16-19, 2009, pp. 901-905.
- [21] Leszak E., Patent US 334 2051A1 published 1967 09-19. Apparatus and Process for Incremental Dieless Forming.
- [22] "SCULPTOR" EU Sixth Framework Program. Contract: 014026.
- [23] J.M. Allwood, H. Utsunomija, A survey of flexible forming processes in Japan. *Int. Journal of Machine Tools & Manufacture*, Vol. 46, pp. 1939-1960, 2006.

# ISOTROPY IN ANY RR PLANAR DYAD UNDER ACTIVE JOINT STIFFNESS REGULATION

Nicola Pio Belfiore\*

Paolo Di Giamberardino\*\*

Imre J. Rudas\*\*\*

Matteo Verotti\*

\* Sapienza University of Rome, Department of Mechanical and Aerospace Engineering

\*\* Sapienza University of Rome, Department of Computer and System Sciences *Antonio Ruberti*

\*\*\* Óbuda University, Rector

## ABSTRACT

The present investigation is dedicated to the study of the static balance at the tip of a planar RR robot. For this case, a configuration can be interpreted, in the *static* sense, as an *isotropic* one when any force applied to the robot wrist yields a small displacement which is theoretically parallel to the applied force (no matter how the force is directed on the plane). This characteristic offers many advantages and it is considered as an optimal design goal. Unfortunately, the conditions to achieve such property in RR manipulators are very restrictive, and until now, only one solution is adopted, with a fixed lengths ratio. The present paper reveals how *any* RR planar robot can achieve isotropy at the tip by using a feedback action at the joints to gain arbitrary elastic coefficients. The new approach of design brings to less restrictive conditions than the previous ones.

Keywords: SCARA, robotic arms, compliance, isotropy

## 1 NOMENCLATURE

$\{s\}$ : wrist position in the Cartesian space (plane);

$\{q\}$ : joint coordinates array;

$[J]$ : geometric Jacobian defined as

$$\{\dot{s}\} = [J]\{\dot{q}\} \quad (1)$$

$\{F_E\}$ : external force;

$\{R_E\}$ : external force mapped in the joint space, namely

$$\{R_E\} = [J]^T \{F_E\} \quad (2)$$

$[K_p]$ : joint passive stiffness matrix  $diag(K_{p1}, K_{p2}, \dots, K_{pn})$  due to the mechanical compliance of the joints components;

$\{R_p\}$ : joint reaction array with

$$\{R_p\} = -[K_p]\{\Delta q\} \quad (3)$$

$[K_c]$ : joint control stiffness matrix, which is assumed in this paper to be  $diag(K_{c1}, K_{c2}, \dots, K_{cn})$ , where the generic element  $K_{c_j}$  of the diagonal represents the feedback gain (in terms of torque) at the corresponding joint  $j$ ;

$\{R_c\}$ : reaction generalized forces due to the control system, arranged into an array, namely,

$$\{R_c\} = -[K_c]\{\Delta q\} \quad (4)$$

$\{F_c\}$ : control reaction force mapped in the Cartesian space, namely

$$\{F_c\} = [J]^{-T} \{R_c\} \quad (5)$$

$\{F\}$ : total active force at the wrist,

$$\{F\} = \{F_E\} + \{F_c\} \quad (6)$$

$[C_p]$ : passive compliance matrix

$$\{\Delta s\} = [C_p]\{F\} \quad (7)$$

due to passive mechanical characteristics of the joint components. Generally speaking, the passive compliance matrix is defined in the Cartesian space, while the joint stiffness matrix refers to the joint space. Therefore  $[C_p] \neq [K_p]^{-1}$ .

$[C_{eq}]$ : equivalent compliance matrix  $\{\Delta s\} = [C_{eq}]\{F_E\}$  that includes the reaction component due to the control systems.

---

Contact author: Nicola Pio Belfiore

Via Eudossiana, 18, 00184 Rome, Italy

Email: belfiore@dma.ing.uniroma1.it

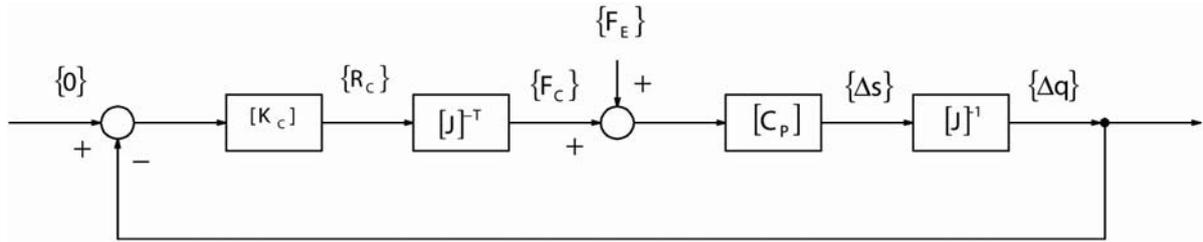


Figure 1 The adopted control scheme.

## 2 INTRODUCTION

RR dyads are very well known in practice and theory, since they have been adopted in many applications. One of them is the well known SCARA robot, which has been used widely in the industry during the last decades. The lengths of the first two links are usually optimized for general purpose. One choice, could be the adoption of an optimum ratio of the first length by the second one, such as, for example,

$$\rho_1 = \frac{l_1}{l_2} = \sqrt{2} . \quad (8)$$

In several cases, as it will be described soon in the paper, this particular ratio is the only one which can guarantee the existence of at least one configuration where isotropy is reached at the wrist. To achieve the same result with a different length ratio, the stiffness at each joint should be varied according to a proper law. However, such modification is rather difficult a task, since the mechanical properties of the hardware should be changed.

The present paper discloses, for the first time in literature, how to get at least one isotropic configuration for any RR planar arm by simply acting on the robot control system.

## 3 REVIEW OF THE PREVIOUS CONTRIBUTION

The concepts of manipulability and isotropy have been considered interesting topics for decades, because they make it easier to improve a manipulator. Actually, they represent a numerical valuation of a certain design and many performance indices can be derived.

Early performance kinematic indices can be found in [1], where the concept of service angle was introduced, and in [2], where the conditioning criterion appeared. Then, some contributions, such as, for example [3-5], introduced the concepts of manipulability and dextrous workspace. Thanks to many other contributions, such as [6-10], the concept of condition number of the Jacobian was introduced. In [11] the kinematic performance was related to the minimum singular value of the Jacobian, as a measure of the distance to singularities. Based on the condition number of the Jacobian, the kinematic

conditioning index was defined as in [10]. Other formulations and various aspects of open chain manipulability have been also investigated in [12-15]. Nearly all of the previous contributions are based on parameterizations of the forward kinematics in local coordinates. In [16-17] manipulability has been formulated under a general mathematical framework, by using the coordinate free methods of Riemann's geometry. Moreover, a great number of cost functions (to characterize the kinematic performances of the manipulator) have been presented in literature. In [18] a cost function to describe kinematic isotropy, manipulability and accuracy has been proposed. In order to remove the configuration dependence from these measures, a new global isotropy index, introduced in [19-20], could compare worst-case values of the workspace as a whole.

## 4 THE INFLUENCE OF JOINT STIFFNESS ON THE RR ISOTROPY MAP

The class of SCARA robots became quite popular because they presented a selective compliance which was used to adjust positions very easily, by using conical insertion procedure before the required operation. Therefore, robots belonging to this class present higher compliance than the others.

The principle of static compliance (or stiffness) is a general topic which is approached firstly by defining the so called joint stiffness matrix  $[K]$ , whose generic element  $i - j$  relates the torques acting on joint  $i$  to the rotation on the joint  $j$ . However, given a serial manipulator, a simple analysis of the mechanical characteristics allows to reach the conclusion that a torque on a joint can not affect the rotation on another joint, unless they are coupled either by a mechanical or by an electronic system.

For this reason, the stiffness matrix in the joint space

$$[K] = \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix} \quad (9)$$

is usually diagonal. Using the simplified notation

$$\begin{aligned} \sin(\vartheta_i) &= s_i & \sin(\vartheta_i + \vartheta_j) &= s_{ij} \\ \cos(\vartheta_i) &= c_i & \cos(\vartheta_i + \vartheta_j) &= c_{ij} \end{aligned} \quad (10)$$

where  $\vartheta_k$  is the  $k$  - joint rotation angle, the compliance matrix can be obtained by means of the relation

$$[C] = [J][K]^{-1}[J]^T \quad (11)$$

where  $[J]$  is the RR Jacobian, namely,

$$[J] = \begin{bmatrix} -l_1 s_1 - l_2 s_{12} & -l_2 s_{12} \\ l_1 c_1 + l_2 c_{12} & l_2 c_{12} \end{bmatrix}. \quad (12)$$

It can be shown easily that for an RR planar dyad,

$$C(1,1) = \frac{(-l_1 s_1 - l_2 s_{12})^2}{k_1} + \frac{l_2^2 s_{12}^2}{k_2}$$

$$C(1,2) = \frac{(-l_1 s_1 - l_2 s_{12})(l_1 c_1 + l_2 c_{12})}{k_1} - \frac{l_2^2 s_{12} c_{12}}{k_2}$$

$$C(2,1) = C(1,2)$$

$$C(2,2) = \frac{(l_1 c_1 + l_2 c_{12})^2}{k_1} + \frac{l_2^2 c_{12}^2}{k_2}$$

Introducing the dimensionless parameters

$$\rho_l = \frac{l_1}{l_2} \quad (13)$$

$$\rho_s = \frac{k_1}{k_2} \quad (14)$$

and setting, with no loss of generality

$$\vartheta_1 = 0, \quad (15)$$

the equations (12) and (11) can be written in the forms

$$[J] = l_2 \begin{bmatrix} -s_2 & -s_2 \\ \rho_l + c_2 & c_2 \end{bmatrix} \quad (16)$$

and

$$[C] = \tilde{c}[\Gamma] \quad (17)$$

respectively, with

$$\tilde{c} = \frac{l_2^2}{k_2} \quad (18)$$

and

$$[\Gamma] = \begin{bmatrix} \frac{s_2^2(1 + \rho_s)}{\rho_s} & -\frac{s_2(\rho_l + c_2 + c_2 \rho_s)}{\rho_s} \\ -\frac{s_2(\rho_l + c_2 + c_2 \rho_s)}{\rho_s} & \frac{\rho_l^2 + 2\rho_l c_2 + c_2^2 + c_2^2 \rho_s}{\rho_s} \end{bmatrix} \quad (19)$$

The condition of parallelism between the external force and the tip displacement imposes a particular structure for  $[C]$  and, consequently, for  $[\Gamma]$ .

In order to obtain such structures, let us introduce a basis in  $\mathfrak{R}^2$ , namely  $(\{v_1\}, \{v_2\})$ ,  $\{v_i\} \in \mathfrak{R}^2$ . Then, a force  $\{F\}$  can be written, with respect to such a basis, as

$$\{F\} = \alpha_1 \{v_1\} + \alpha_2 \{v_2\} = [V]\{\alpha\} \quad (20)$$

with

$$[V] = [\{v_1\} \quad \{v_2\}]$$

$$\{\alpha\} = \{\alpha_1 \quad \alpha_2\}^T$$

Changing the force means changing the coefficient vector  $\{\alpha\}$ , since a biunique relationship between them holds.

Now, the parallelism condition holds once that

$$\{\Delta s\} = \lambda \{F\} = \lambda [V]\{\alpha\} \quad (21)$$

Making use of (20) and (21), the relationship between force and displacement involving  $[C]$  can be written as

$$\lambda [V]\{\alpha\} = [C][V]\{\alpha\}$$

or, equivalently, after some computations,

$$(\lambda [I] - [V]^{-1}[C][V])\{\alpha\} = \{0\} \quad (22)$$

Condition (22) must hold for any  $\{\alpha\} \in \mathfrak{R}^2$ . It implies that, necessarily,

$$\lambda [I] - [V]^{-1}[C][V] = [0]$$

from which one gets

$$[C] = \lambda [V][I][V]^{-1} = \lambda [I]$$

Hence,  $[C]$ , and then  $[\Gamma]$ , must be diagonal with the same values for each entry of the diagonal and so the conditions under which parallelism, i.e. isotropy, is achieved can be written as

$$C(1,1) - C(2,2) = 0 \quad (23)$$

$$C(2,1) = C(1,2) = 0 \quad (24)$$

yielding

$$-\frac{2c_2^2 + 2c_2^2 \rho_s + 2\rho_l c_2 + \rho_l^2 - \rho_s - 1}{\rho_s} = 0 \quad (25)$$

$$-\frac{s_2(\rho_l + c_2 + c_2 \rho_s)}{\rho_s} = 0 \quad (26)$$

whose non critical solutions can easily obtained in the closed form

$$\rho_s = \rho_l^2 - 1 \quad (27)$$

$$g_2 = \pi - \arccos\left(\frac{1}{\rho_l}\right). \quad (28)$$

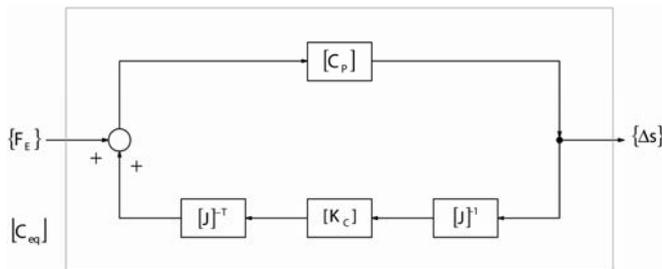


Figure 2 The equivalent compliance matrix.

## 5 THE CONTROL ACTION FOR THE ARBITRARY RR DYAD STIFFNESS SETTING

As stated in the previous paragraphs, in this paper the overall stiffness of the robot is evaluated by taking into account the following hypotheses:

- the links are perfectly rigid bodies;
- there is a passive compliance at the joints due to the mechanical characteristics of the revolute system, including the speed reducers; for example, this assumption holds for any SCARA architecture, where the Harmonic Drives are used to have a certain amount of compliance, as they were used to adjust positioning operation in the plane;
- there is also an active compliance that can be provided by a feedback control scheme.

It is well known that any robotic structure is provided by a control system that guarantees joint control, allowing to get any (feasible) configuration. Usually, such a control is designed in order to have zero steady state error, that is the desired joint configuration should be reached exactly and any disturbance force or torque should be compensated.

In the present case, a small displacement not only is acceptable but must be present, and in a steady state it should be proportional to the external action that the control system considers as a disturbance. This means that the normal action of the control system must be modified.

There are two simple ways to get the desired behaviour. The former is the inhibition of the integral action of the controller, without which the zero steady state error is no longer obtained. The latter makes use of an additional feedback with a simple proportional control. In both cases, under the hypothesis that the condition under which the isotropy is studied, i.e. that the mechanical structure is fixed in a desired position by the action of the embedded control system, so having zero velocities and accelerations,

all the dynamic behaviour can be neglected and the whole control scheme can be simplified as in Figure 1.

With reference to the Nomenclature,  $\{\Delta q\}$  and  $\{\Delta s\}$  are small displacement arrays in the joint and Cartesian spaces. From the hypotheses, the control acts in order to keep the reached configuration  $\{\hat{q}\}$  stable, and therefore the desired displacement is null. If an external force  $\{F_E\}$  acts on the tip, a displacement  $\{\Delta s\}$  will arise. Such displacement in the Cartesian space can be measured at the joints, as a displacement array in the joint space  $\{\Delta q\}$ . The latter is used as reaction signal, from which a reaction force array  $\{R_C\}$  is generated by the control system. In the Cartesian plane  $\{R_C\}$  is mapped on the vector  $\{F_C\}$  which acts on the tip of the second link as well as the external applied force  $\{F_E\}$ . The tip displacement in the Cartesian plane is due therefore to both the contribution and can be calculated by means of the compliance matrix  $[C_p]$ , only due to the passive mechanical compliance, that has to be evaluated in the desired configuration  $\{\hat{q}\}$ .

Finally, the displacements in the joint space can be obtained by means of (1).

With reference to the Nomenclature, since the system must be in static balance we have

$$\{F_E\} + \{F_C\} + [J]^T \{R_p\} = \{0\} \quad (29)$$

From (3) and (1)

$$\{F_E\} + \{F_C\} = [J]^T [K_p] [J] \{\Delta s\} \quad (30)$$

By inverting the general expression (11), the inverse of  $[C_p]$  is actually equal to

$$[C_p]^{-1} = [J]^T [K_p] [J] \quad (31)$$

and so (30) can be written immediately as

$$\{F_E\} + \{F_C\} = [C_p]^{-1} \{\Delta s\} \quad (32)$$

This relation justifies the control scheme, depicted in Figure 1 as the transposition of the static balance of the robot subject to the action of the external force and of the active control torques.

## 6 THE EFFECT OF THE ADOPTION OF THE DESCRIBED CONTROL SYSTEM ON THE APPARENT COMPLIANCE MATRIX

If we consider the control system as an integrated part of the whole robot, the static behaviour of the manipulator appears different from the pure passive mechanical system.

The situation can be represented as in Figure 2, where the external force  $\{F_E\}$  is regarded as the only active (external) force applied on the robot tip, so excluding the control contribution. According to this scheme, the consequent displacement is

$$\{\Delta s\} = [C_{eq}] \{F_E\} \quad (33)$$

where  $[C_{eq}]$  is the equivalent compliance matrix, which takes into account the contribution of the reacting capacity of the control system as a *reaction* generalized force.

By substituting the expression of the force

$$\{F_E\} = [C_{eq}]^{-1} \{\Delta s\} = [C_{eq}]^{-1} [J] \{\Delta q\} \quad (34)$$

into (29), and taking into account the equations (3), (4) and (5), the relation

$$[C_{eq}]^{-1} [J] \{\Delta q\} - [J]^T [K_c] \{\Delta q\} = [J]^T [K_p] \{\Delta q\} \quad (35)$$

is obtained, from which, after little algebra, the equivalent compliance matrix

$$[C_{eq}] = [J] ([K_c] + [K_p])^{-1} [J]^T \quad (36)$$

can be deduced.

This relation shows that by using the action of a control system like the one assumed in this paper the overall stiffness matrix changed by simply adding the control joint stiffness matrix to the passive joint stiffness matrix

$$[K_{eq}] = [K_c] + [K_p] \quad (37)$$

While  $[K_p]$  depends only from the mechanical components, the matrix  $[K_c]$  can be easily modulated via software.

## 7 THE PROPOSED ALGORITHM

In the isotropic configuration,  $[C]$  is diagonal. In particular, using the notation introduced in (17) and (18),  $[\Gamma] = [I]$  is the identity matrix, while  $\tilde{c}$  corresponds to the eigenvalue of the compliant matrix. Furthermore, considering (27), (28) and (37), the joint control stiffness are

$$k_{c1} = \frac{l_1^2 - l_2^2}{\tilde{c}} - k_{p1} \quad (38)$$

$$k_{c2} = \frac{l_2^2}{\tilde{c}} - k_{p2} \quad (39)$$

According to the control scheme depicted in Figures 1 and 2, the following algorithm can be adopted in order to

achieve isotropy in a given RR planar robot which has any set of link lengths and passive joint stiffness

- 1) acquire the link lengths  $l_1$  and  $l_2$ ; ratio  $\rho_l$  is therefore a fixed value; actually, such lengths are those of the robot which the control system is to be dedicated to;
- 2) measure the passive compliance coefficients  $k_{p1}$  and  $k_{p2}$  at the two joints;
- 3) chose the desired overall compliance  $\tilde{c}$  at the tip, as a scalar number (it has the dimension of a displacement by a force, and therefore  $[m \cdot N^{-1}]$ );
- 4) impose, via software, the gains  $k_{c1}$  and  $k_{c2}$  on the two joints, as in (38) and (39).

Isotropy can be granted if (27) and (28) are both satisfied.

The advantage of including active compliance control in parallel to the natural mechanical compliance at the joint components are:

- the overall compliance  $\tilde{c}$  can be selected via software;
- isotropy can be achieved for any link ratio  $\rho_l = \frac{l_1}{l_2}$ .

The main drawbacks are:

- to achieve isotropy (28) must be satisfied;
- this means that isotropy can be obtained only for those positions for which the robot tip is positioned on a circle that has its centre in the first revolute joint centre, and radius  $r = \sqrt{l_1^2 - l_2^2}$ .

For example, if  $k_{p1} = k_{p2}$ ,  $l_1 = 1$  and  $l_2 = \frac{\sqrt{2}}{2}$ , as in the

SCARA robots, then  $\rho_l = \sqrt{2}$  and  $\rho_s = 1$ , and so equation (27) is already satisfied with no need of the control action. In such case, the scheme is no more the one depicted in Figure 1, because the feedback is taken on the joint servo actuator before the rotation due to the mechanical passive compliance. Instead, than (33), (7) holds alone, being  $\{F_c\} = \{0\}$ . Of course, in such a case, from (28), isotropy is achieved only if  $\theta_2 = \pm 135^\circ$ .

## 8 EXAMPLES OF APPLICATION

By assigning a unit length to the first link and an arbitrary length to the second one, as for example,  $l_2 = 0.65$  m and  $l_2 = 0.90$  m, and choosing the overall compliance  $\tilde{c} = 0.001$  mN<sup>-1</sup>, the algorithm described in the previous paragraph can be adopted literally.

The results are summarized in Tables I and II.

Finally, Figures 3 and 4 show the simulated compliance ellipse for the two manipulators, which becomes a circle, for the assigned values of joint control stiffness.

Table I

$k_p$ [Nm]	$k_c$ [Nm]	$k$ [Nm]	$l$ [m]	$\mathcal{G}$ [deg]
$k_{p1} = 20.0$	$k_{c1} = 557.5$	$k_1 = 577.5$	$l_1 = 1.0$	$\mathcal{G}_1 = 0$
$k_{p2} = 20.0$	$k_{c2} = 402.5$	$k_2 = 422.5$	$l_2 = 0.65$	$\mathcal{G}_2 = 130.54$

Table II

$k_p$ [Nm]	$k_c$ [Nm]	$k$ [Nm]	$l$ [m]	$\mathcal{G}$ [deg]
$k_{p1} = 30.0$	$k_{c1} = 160.0$	$k_1 = 190.0$	$l_1 = 1.0$	$\mathcal{G}_1 = 0$
$k_{p2} = 25.0$	$k_{c2} = 785.0$	$k_2 = 810.0$	$l_2 = 0.90$	$\mathcal{G}_2 = 154.16$

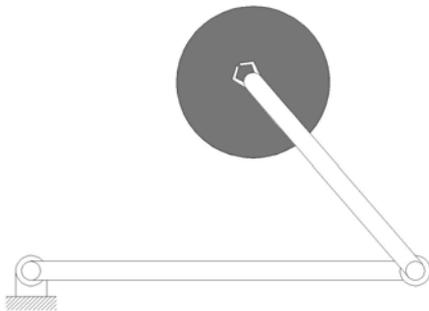


Figure 3 The manipulator described in Table I and the compliance ellipse in the isotropic configuration.

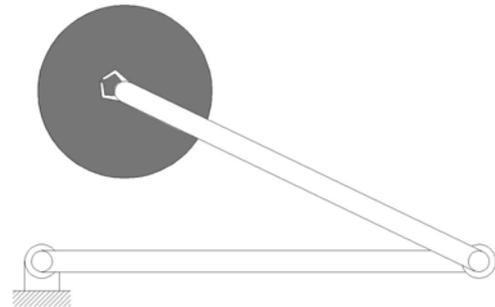


Figure 4 The manipulator described in Table II and the compliance ellipse in the isotropic configuration.

## 9 CONCLUSIONS

The arguments presented in this paper have shown how it is possible to establish, for any RR planar robot, one circular arc in the working plane where static isotropy is achieved. This result has been obtained by using a new control system that acts in parallel to the natural passive compliance of the joint mechanical components. Such method allows to modify the overall elastic behaviour of the given manipulator in order to get isotropy in the Cartesian plane also if such robot would never be able to do it, for its own set of link lengths and mechanical joint stiffness. For this reason, the suggested design methodology enlarges the designer capacity in getting this optimal goal more easily.

## REFERENCES

- [1] Vinogradov I. B., Kobrinski A. E., Stepanenko Y. E. and Tives L. T., Details of kinematics of manipulators with the method of volumes. *Mekhanika Mashin*, No. 27-28, pp. 5-16, 1971.
- [2] Yang D. C. H. and Lai Z. C., On the Dexterity of robotic manipulators - service angle. *ASME Journal of Mechanisms, Transmissions, and Automation in Design*, Vol. 107, pp. 262-270, 1985.
- [3] Kumar, A. and Waldron K. J., The workspaces of a mechanical manipulator. *ASME, Design Engineering Technical Conference*, Beverly Hills (U.S.A.), ASME Paper No. 80-DET-107, 1980.
- [4] Yoshikawa T., Analysis and control of robot manipulators with redundancy. *Robotics Research: The First International Symposium*, 735-747, 1984.
- [5] Yoshikawa T., Manipulability of robotic mechanisms. *The International Journal of Robotics Research*, Vol. 4, No. 2, pp. 3-9, 1985.
- [6] Paul R.P. and Stevenson C.N., Kinematics of robot wrists. *The International Journal of Robotics Research*, Vol. 2 No.1, pp. 31-38, 1983.
- [7] Forsythe G.E. and Moler C.B., *Computer solution of linear algebraic systems*. Prentice-Hall Englewood Cliffs, NJ, 1967.
- [8] Salisbury J.K. and Craig J.J., Articulated hands: force control and kinematic issues. *The International Journal of Robotics Research*, Vol. 1, No. 1, pp. 4-17, 1982.
- [9] Angeles J. and Rojas A., Manipulator inverse kinematics via condition number minimization and continuation. *International Journal of Robotics and Automation*, Vol. 2, No. 2, pp. 61-69, 1987.
- [10] Angeles J. and Lopez-Cajun C.S., Kinematic isotropy and the conditioning index of serial robotic

- manipulators. *The International Journal of Robotics Research*, Vol. 11, No. 6, pp. 560-571, 1992.
- [11] Klein C.A. and Blaho B.E., Dexterity measures for the design and control of kinematically redundant manipulators. *The International Journal of Robotics Research*, Vol. 6, No. 2, pp. 72-83, 1987.
- [12] Gosselin C., Dexterity indices for planar and spatial robotic manipulators. *Proceedings of the IEEE International Conference on Robotics and Automation*, Cincinnati, (U.S.A.), Vol. 1, pp. 650-655, 1990.
- [13] Gosselin C. and Angeles J., A global performance index for the kinematic optimization of robotic manipulators. *ASME Journal of Mechanical Design*, Vol. 113, No. 3, pp. 220-226, 1991.
- [14] Klein C.A. and Miklos T.A., Spatial robotic isotropy. *The International Journal of Robotics Research*, Vol. 10, No. 4, pp. 426-437, 1991.
- [15] Angeles J., The design of isotropic manipulator architectures in the presence of redundancies. *The International Journal of Robotics Research*, Vol. 11, No. 3, pp. 196-201, 1992.
- [16] Park F.C. and Brockett R.W., Kinematic dexterity of robotic mechanisms. *The International Journal of Robotics Research*, Vol. 13, No. 1, pp. 1-15, 1994.
- [17] Park F.C., Optimal robot design and differential geometry. *Journal of Mechanical Design*, 117:87, 1995.
- [18] Kim, J. O. and Khosla, P. K., Dexterity measures for design and control of manipulators, *Proceedings of IEEE International Workshop Intelligent Robot Systems*, Osaka, Japan, pp. 758-763, 1991.
- [19] Stocco L., Salcudean S.E. and Sassani F., Mechanism design for global isotropy with applications to haptic interfaces. *ASME Winter Annual Meeting*, Dallas, U.S.A., pp. 115-122, 1997.
- [20] Stocco L., Salcudean S.E. and Sassani F., Fast constrained global minimax optimization of robot parameters. *Robotica*, Vol. 16, No. 6, pp. 595-605, 1998.



# CAD BASED TECHNIQUES FOR WORKSPACE ANALYSIS AND REPRESENTATION OF THE 3-CRS PARALLEL MANIPULATOR

Khaled Assad Arrouk\*

Belhassen Chedli Bouzgarrou\*\*

Grigore Gogu\*\*

\* Clermont Université, Université Blaise Pascal, EA 3867, Laboratoire de Mécanique et Ingénieries,  
BP 10448, F-63000 Clermont-Ferrand

\*\* Clermont Université, IFMA, EA 3867, Laboratoire de Mécanique et Ingénieries,  
BP 10448, F-63000 Clermont-Ferrand

## ABSTRACT

This paper presents a set of techniques, based on a geometrical approach, for workspace determination and analysis. These techniques are applied for design study of a novel modular parallel manipulator, 3-CRS, with 6-DOF. The proposed method is implemented in the CATIA CAD environment, which provides powerful tools for graphical programming and geometric feature handling. A new type of workspace representation is introduced allowing a fine simultaneous characterization of the translation and orientation capacities of the mobile platform.

Keywords: Parallel robots, workspace, CAD, joint limits, optimization

## 1 INTRODUCTION

Workspace determination and analysis play an essential role in the synthesis and design of parallel robotic manipulators (PRMs). The workspace can be seen as a performance criterion. Indeed, its volume, for different platform orientations, is used to measure the dexterity properties [1]. The problem of workspace determination of PRMs has been addressed by several research studies in order to verify the accessibility to the desired poses according to different robotic applications [2-5]. Nevertheless, workspace characterization problem is a challenging issue since kinematic equations of PRMs are rather complex in general.

For spatial 6-DOF PRMs, the impossibility of graphical representation of the full six-dimensional workspace has made extremely complicated the understanding of the manipulator characteristics. Thus, a graphical representation of the workspace will be possible only if 3-n parameters are fixed, where n is the manipulator degree of

freedom. Consequently, the two main types of workspace are the position (or translational) workspace and the orientation (or rotational) workspace. According to several possible three-dimensional graphical representations, and depending on the constraints that can be imposed to the parameters, various definition of workspace can be obtained: the constant orientation workspace, the total orientation workspace, the maximal or reachable workspace, the inclusive workspace, the dexterous workspace which is a special case of the total orientation workspace [6]. Several techniques are used for workspace determination and representation. The numerical discretization methods are largely used and easy to implement. They have many advantages: discretization is usually simple to implement and all kinematic constraints can be taken into account since they can be easily verified for a given pose. Several applications of these methods can be found in [8-10]. However, they are computationally intensive. In fact, these methods are based on operational space grid and computation time increases exponentially with the mesh resolution. Furthermore, they introduce uncertainty related to the mesh density [7]. Consequently, relatively approximate data is given about the exact boundaries of the workspaces. The algebraic methods by discretization use the direct and inverse geometrical models to calculate the set of configurations that the manipulator

---

Contact author: Khaled Assad Arrouk<sup>1</sup>,  
Belhassen Chedli Bouzgarrou<sup>2</sup>, Grigore Gogu<sup>3</sup>

<sup>1</sup>E-mail: khaled.arrouk@ifma.fr

<sup>2</sup>E-mail: belhassen-chedli.bouzgarrou@ifma.fr

<sup>3</sup>E-mail: grigore.gogu@ifma.fr

can reach. Quadtree and Octrees methods can be cited as examples. They offer many advantages over the discretization methods [11]: less memory space is required, particularly well adapted to the representation of the complex forms of the three-dimension PRMs workspace [12] and implicit adjacency graph, which facilitates the analysis of workspace connectivity. But this depends on the realization of the octree, which must be neither too fine, this causes the appearing of the discontinuities, nor too large, which on the contrary eliminates discontinuities that may exist in reality. The geometrical methods have been used efficiently for representing the workspace of parallel robots. The geometric approach allows establishing the nature of the boundary of the workspace. The principle of these approaches is to deduce, from the constraints on each kinematic chain (limb) separately, a geometrical entity which describes all the possible locations of the end-effector that satisfy the leg constraints. Then, the robotic manipulator workspace is constituted by the intersection of all the entities of the kinematic chains [2]. This technique was used for the calculation of the workspace of a Gough platform when its orientation is fixed (constant orientation or translational workspace) [13, 14]. Merlet [6] has pointed out that the geometrical approaches are very efficient for the determination of various types of workspace. The inverse geometric model was used to determine the boundaries workspace by this technique [13, 15]. The main contribution of this paper is to extend the geometrical approach to determine, analyze, and optimize the workspace of new 6-DOFs PRM architecture: 3-CRS. In addition, a new type of workspace representation is proposed. This method pertains to a geometrical approach and emphasizes on the use of CAD environment as natural framework for developing such approach.

## 2 PRESENTATION OF THE 3-CRS MANIPULATOR

The 3-CRS PRM comprises three identical kinematic chains connecting the fixed base and the moving platform. The kinematic chains are described by abbreviation of the joint types, starting from the fixed base and ending at the moving platform. Each limb is connected to the mobile platform by a passive spherical joint. On the other extremity of the limb there is an actuated prismatic joint followed by an actuated revolute joint. These two joints have the same direction/axis. Thus this pair of joints is equivalent to an actuated cylindrical joint. Finally, a second passive revolute joint is perpendicularly oriented relatively to the first actuated revolute joint axis. Figure 1 shows the CAD model of 3-CRS PRM and its kinematic scheme. Figure 2 shows the graph of joints of 3-CRS robot.

### 2.1 STRUCTURAL ANALYSIS OF 3-CRS ROBOT

The degree of mobility  $M$  is the number of independent motions of a mechanism. It can be deduced from the relation (1), based on the new formulae developed by G.

Gogu [16]. These formulae are based on the linear transformations theory and given by:

$$M = \sum_{j=1}^N f_j - r \quad (1)$$

Where  $N$  is the number of joints and  $\sum_{j=1}^N f_j$  is the number of independent motion parameters of the joints before the loop closures ( $f_j$  is the  $j^{\text{th}}$  joint mobility).  $r$  is the number of joint parameters losing their independence after loop closures. For a robot having  $k$  simple limbs,  $r$  is given by the relation (2):

$$r = \sum_{i=1}^k S_{Ai} - S_C \quad (2)$$

Where  $S_{Ai}$  is the connectivity of the  $i^{\text{th}}$  limb isolated from the parallel mechanism and  $S_C$  is the connectivity of the mobile platform in the parallel mechanism. Connectivity of a limb  $i$  is given by the dimension of its operational vector space  $R_{Ai}$ :

$$S_{Ai} = \dim(R_{Ai}) \quad (3)$$

The connectivity of the mobile platform is given by:

$$S_C = \dim\left(\bigcap_{i=1}^k R_{Ai}\right) \quad (4)$$

The 3-CRS manipulator has 3 revolute joints, 3 cylindrical joints and 3 spherical joints. Therefore, we have:

$$\sum_{j=1}^{12} f_j = 18, \quad \sum_{i=1}^3 S_{Ai} = 18, \quad S_C = 6$$

Finally, by applying relations (1-4), the mobility of 3-CRS robot is then equal to 6.

### 2.2 MODULAR DESIGN AND RECONFIGURABILITY

The 3-CRS is a modular parallel architecture. It can be assembled according to infinity of configurations defined by the cylindrical joint orientation of each limb. Design study can be carried out in order to adapt geometric configuration to task requirements. Figure 3 shows some particular forms of the fixed base that can be used to reconfigure the 3-CRS robot. The slider guides can be arranged in different ways: horizontal star shape, star shape with inclination  $\varphi$  with respect to the horizontal plane, vertical star shape with  $\varphi = 90^\circ$ . In Figure 3, constant orientation workspaces corresponding to each configuration are represented.

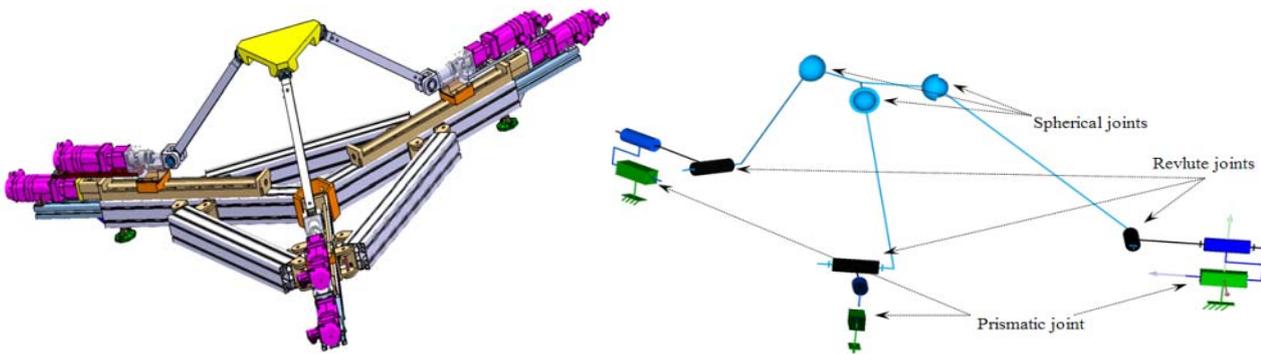


Figure 1 3-CRS CAD model and its kinematic scheme.

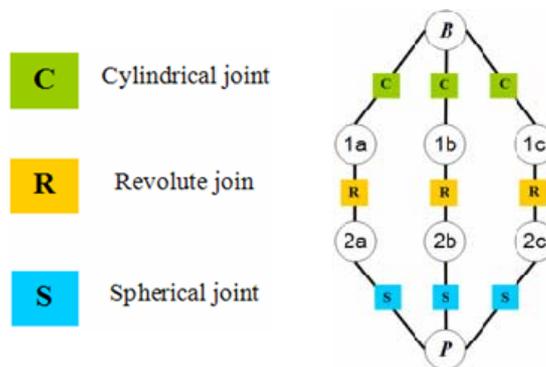


Figure 2 Graph of joints of the 3-CRS manipulator.

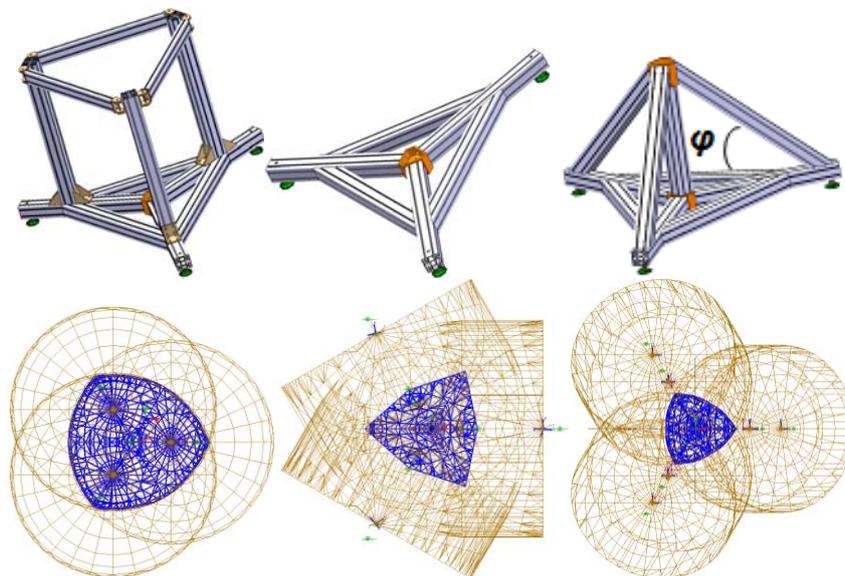


Figure 3 Different forms of the fixed base and their corresponding.

### 3 WORKSPACE ANALYSES

#### 3.1 CONSTANT ORIENTATION WORKSPACE

There are several definitions of the workspace. It can be defined as the positions and rotations accessible by end-effector characteristic point (EECP), which is generally

located on the mobile platform [17]. The equations describing the workspace are usually extremely complex to be solved by using the traditional techniques. Thus, the CAD geometrical approach is used in this paper to determine the workspace of 3-CRS robot. CAD software such as CATIA provides powerful tools for graphical

programming and geometric feature handling. Therefore, the implementation of this approach in a CAD environment becomes implicitly and naturally imposed.

The approach presented in this paper demonstrates that CAD tools could be a serious alternative, relevant and effective to the complex mathematical resolution of workspace representation. We can produce the workspace as a wireframe and even solid, faster than a complex method such as the algebraic or numerical methods. In discretization method, which gives only an approximation of the shape of the workspace even when a rather fine step is used, it is very difficult to obtain a good quality representation of the workspace without additional graphical manipulation. By embedding the problem of workspace determination within a CAD framework, these methods will become more accessible to industry and more accurate.

In a first step, the proposed procedure for constant orientation workspace determination of the 3-CRS robot consists in considering all limbs as independent serial chains having the mobile platform as end-effector. Then, the volume swept by the EECP of each chain is determined for a given orientation of the end-effector. Vertex spaces (the volumes can be reached by the extremities of the limbs) are shown in Figure 4(a). They are displaced by an offset vector OC presented in Figure 4(b) which is defined by the vector the attachment point of the moving platform and the centre of this platform.

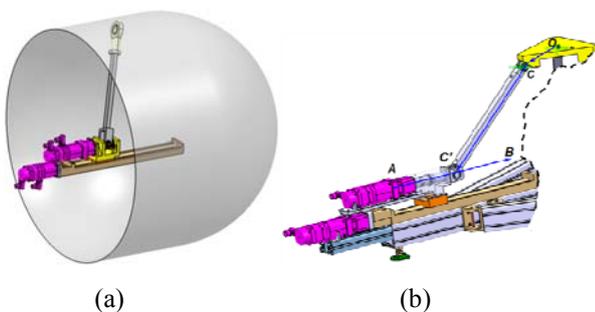


Figure 4 Volume reached by limb (a), robot parameterized model (b).

It is assumed here that there are no limits in the passive revolutes and spherical joints.

In order to take into account the loop closure constraints, the constant orientation workspace is then obtained by the intersection of three volumes. Figure 5 shows the three vertex spaces attained by the three limbs. The intersection between these volumes is considered for the horizontal star configuration of the fixed base. Only the portion superior to the base plane is represented. The geometrical dimensions of the 3-CRS robot with the workspace presented in Figure 5 as follows: linear stork of the cylindrical joint given by the distance AB = 438 mm, the position of the spherical joint attachment point relative to the platform centre given by the distance OC = 200 mm, the distance between the centre of the fixed base and the initial position of the

cylindrical joint is 689 mm, finally the length of the upper link given by the distance C'C = 590 mm.

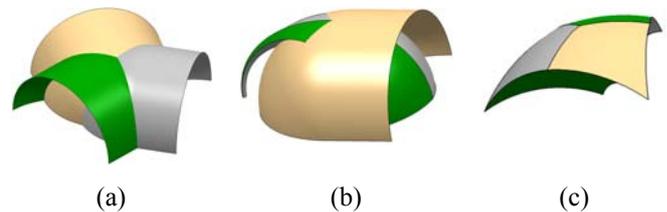


Figure 5 Intersection three upper volumes reached by the three limbs (a), the first intersection Boolean operation (b), the second intersection Boolean operation corresponds to the constant orientation workspace (c).

### 3.2 OPTIMIZATION PROBLEM

The design of robotic manipulators can be based on several performance criteria: dexterity, rigidity, precision, etc. Since workspace characterizes the potential robot utilization, it is a crucial issue in the design process. Thereafter, workspace volume can be adopted as a performance index to perform design optimization. It depends on the robot geometrical design variables. Design optimization problem can be resumed as following:

$$\begin{cases} \text{maximize } W(\xi) \\ \xi_{i,\min} \leq \xi_i \leq \xi_{i,\max} \end{cases} \quad (5)$$

Where  $W$  is a constant orientation workspace volume and  $\xi$  is the array of design variables constrained to vary in a given range. The maximization of the workspace volume aims to extend the capacities of the robot. Many parameters influence the volume and the shape of the constant orientation workspace. The parameters shown in Figure 6 include design variables (distances OC, AB, C'C), the inclination angle  $\varphi$  and the platform orientation variables, and the distance T between the fixed base centre and the centre of the cylindrical joint in the closest position to the fixed point centre. Platform orientation is parameterized by using Euler angles convention: the first rotation is about the x-axis (roll)  $\alpha$ , followed by the rotation about y-axis (pitch)  $\beta$ , and finally a rotation about z-axis (yaw)  $\gamma$ . We consider here the constant orientation workspace for orientation parameters  $\alpha=\beta=\gamma=0^\circ$ .

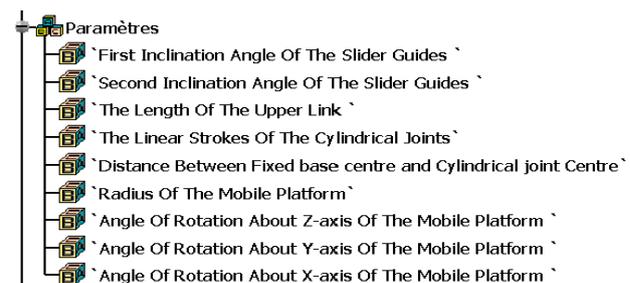


Figure 6 3-CRS workspace parameterization.

### 3.3 DESIGN VARIABLE ANALYSIS

This analysis is illustrated for various design variables. A particular emphasis can be given to the inclination angle  $\varphi$ , see Figure 3. This angle induces changes of the geometric model equations of the 3-CRS robot. Several values of  $\varphi$  have been tested while keeping all others design variables constant as mentioned above. By examining Figure 7, it can be concluded that the volume of the constant orientation workspace increases clearly from  $\varphi = 0^\circ$  to  $\varphi = 30^\circ$  and decreases shapely for superior values of  $\varphi = 30^\circ$ .

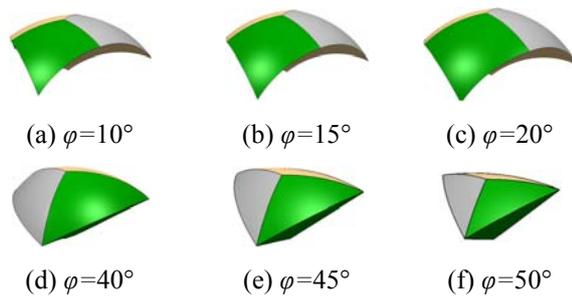


Figure 7 3-CRS workspace for different inclination of the slider guides of the prismatic joints.

The geometrical optimization implemented in CATIA CAD tool aims to maximize the workspace volume as function of the design parameters listed in Table I. The initial design variables correspond to the workspace presented in Figure 8(a) with a volume of  $W=74.2\text{dm}^3$ . A first optimization is performed where the linear stroke of the cylindrical joints is limited by  $V_{\min}=450\text{mm}$  and  $V_{\max}=1000\text{mm}$ . The optimal values of the design variables are given in Table I. and correspond to the workspace presented in Figure 8(b), with a volume of  $W=96.2\text{dm}^3$ . In the second optimisation, the linear stroke of the cylindrical joints is limited by  $V_{\min}=450\text{mm}$  and  $V_{\max}=2000\text{mm}$ . Figure 8(c) shows the optimal constant orientation workspace obtained in this optimization with a volume  $W= 220.5\text{dm}^3$  with the optimal values of the design variables are given in Table I.

TABLE I - Optimization results

Design variables	Initial values	1 <sup>st</sup> Optimization	2 <sup>nd</sup> Optimisation
OC (mm)	150	500	155.4
C'C mm)	590	1000	1000
AB (mm)	438	1000	2000
$\varphi$ (°)	25	32.6	0
T (mm)	250	342.2	55.9
$W$ (dm <sup>3</sup> )	74.2	96.2	220.5

The optimization process is based on the simulated annealing algorithm. The number of iterations realized to reach the objective is 2380 for the first optimisation and 1006 in the second optimization. The time of these optimizations is about 60 minutes. They are performed on a computer which has the following hardware characteristics: CPU 3.20 GHz, 2.0 GByte.

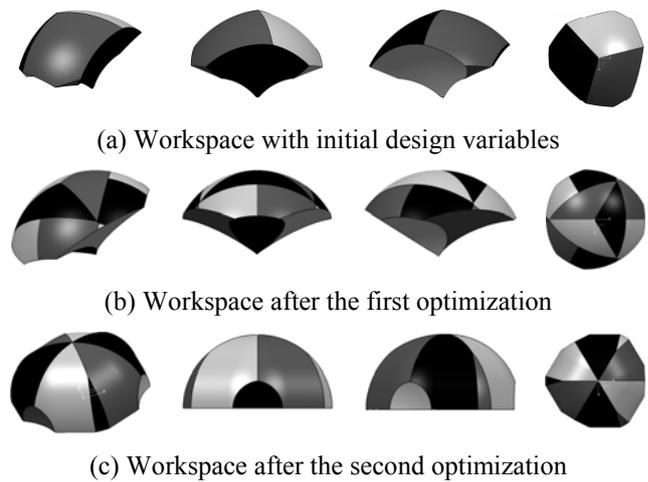


Figure 8 Workspace optimization.

### 3.4 INFLUENCE OF THE JOINT LIMITS

The proposed geometrical approach is also available when other complex joint limits are considered. In the previous analyses, the linear stroke of the actuated cylindrical joints has been implicitly considered as a design variable. However, the range of motion of the passive spherical joints attached to the mobile platform is restricted in reality due to the choice of standard commercial component. This joint limitation restricts the relative orientation between the mobile platform and the link CC'. This link must be within a cone volume delimited by a spherical surface as shown in Figure 9 and 10. The attainable region of the point C, see Figure 4(b) belongs to a spherical cap shown in Figure 9, its radius equals to the length of the upper link (distance C'C in Figure 4(b)) and its centre is the point C'. In order to generate the volume swept by the EECF of each chain, the spherical surface (blue surface) will be extruded along the vector A'A which represents the linear stroke of the cylindrical joints as it is shown in Figure 10.

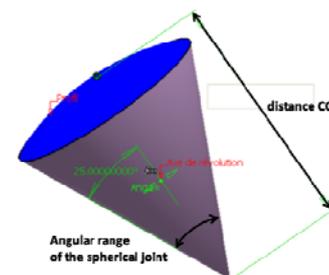


Figure 9 Spherical cap represents the region attached by the upper link.

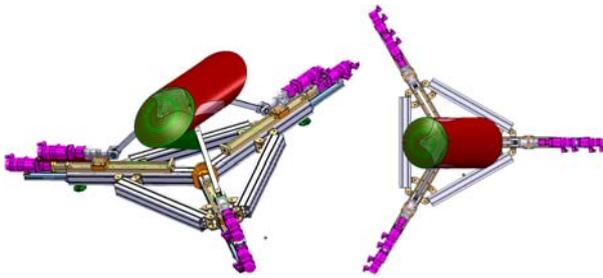
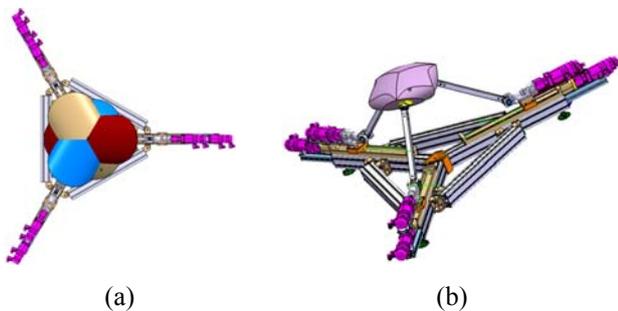


Figure 10 Extrusion of a spherical surface in the direction of the linear stroke of the cylindrical joint.

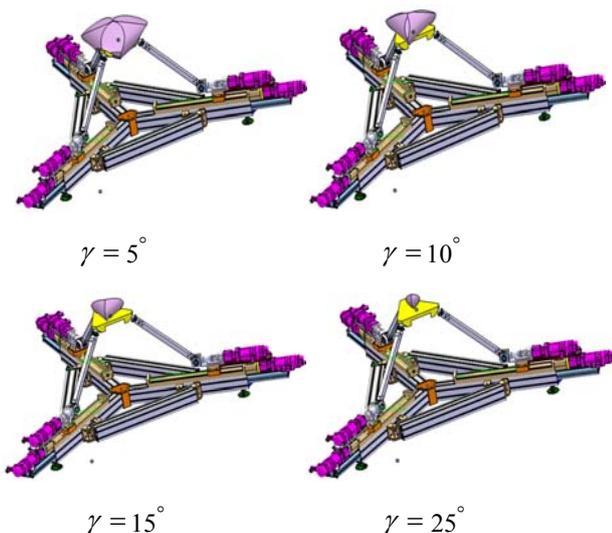
Figure 11(a) shows the three vertex volumes reached by the extremities of the three limbs. After that, two successive intersection Boolean operations will be applied to determine the common volume which represents the constant orientation workspace of 3-CRS as it shown in Figure 11(b).



(a) (b)

Figure 11 Three vertex volumes (a), constant orientation workspace (b).

Figure 12 depicts the influence of the orientation of the mobile platform around its Z-axis (yaw angle  $\gamma$  for the following angles:  $5^\circ$ ,  $10^\circ$ ,  $15^\circ$  and  $25^\circ$  respectively). It can be concluded that the volume of the constant orientation workspace of 3-CRS robot decreases sharply when the yaw angle of the mobile platform increases.



$\gamma = 5^\circ$

$\gamma = 10^\circ$

$\gamma = 15^\circ$

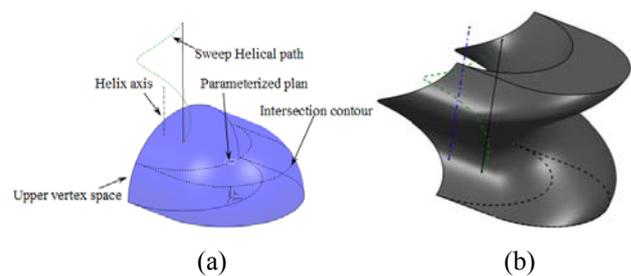
$\gamma = 25^\circ$

Figure 12. Yaw angle influence on the constant orientation workspace.

#### 4 NEW WORKSPACE REPRESENTATION

For PRMs having more than three degrees of freedom it is not possible to have graphical representations of their full workspace. Thus, projected representations are used by fixing  $n-3$  parameters. Accordingly, in all the actual researches, the PRMs workspaces are frequently determined by the 3-D constant orientation workspace, and the 3-D orientation workspace. In this section, a new representation useful to analysis and characterize the workspace is proposed. This method is based on a new 3-D mixed representation of workspace with two degrees of translation and one degree of rotation, i.e. the 6-DOFs 3-CRS PRM is considered as 3-DOFs planar parallel robot. Translations are considered in the plane of the mobile platform and rotation is about the normal to this plane. This is obtained when the two first orientation parameters ( $\alpha$  and  $\beta$ ) are fixed as well as the distance  $d$  from the platform plane to the origin.

The algorithm used here is implemented in a CAD environment and consists in the following steps: firstly, a parameterized plane  $\mathcal{P}$  with two rotations and one distance  $d$  is created. We consider the translational workspace in this plane. Then, the intersection surfaces between this plan and the upper vertex spaces are constructed as it is shown in Figure 13(a). Once the three closed contours delimiting three parameterized surfaces are identified, sweeping operations are applied along helical paths. These helical curves are constructed around the axis normal direction  $N$  to the plane  $\mathcal{P}$ , as illustrated in Figure 13(b). These swept volumes correspond to three-dimensional “potential” workspace (two positions and one orientation) of each limb considered separately having the mobile platform as end-effector (see Figure 14).



(a)

(b)

Figure 13 Components to construct swept volume (a), swept volume.

Finally, intersection Boolean operations are applied to obtain the common volume corresponding to a three-dimensional workspace for the 3-CRS manipulator. This volume, Figure 15, describes the ability of the mobile platform to realize to translate in a plane and spinning rotation around the normal to this plane.

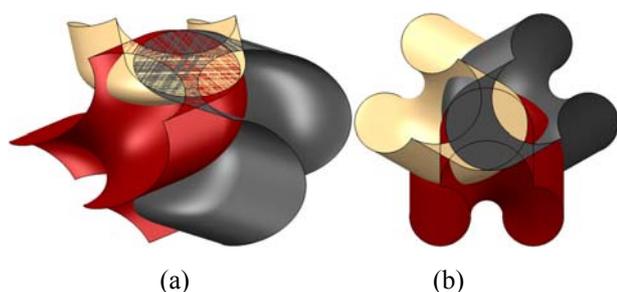


Figure 14 Three swept volumes in 3D (a), top view (b).

Quantitative insight within the whole workspace of 3-CRS can be obtained by cross-section at various altitudes  $\theta_4 > \theta_3 > \theta_2 > \theta_1$ . Figure 15 (a) illustrates the influence of the spin angle of the mobile platform on the constant orientation workspace. It can be concluded from this figure that the constant orientation workspace decreases when the orientation of the mobile platform increases. Figure 15 (b) shows the workspace for several altitudes of the parameterized plane which confirms that the capacity in translation and orientations decreases when the altitude of the plane increases.

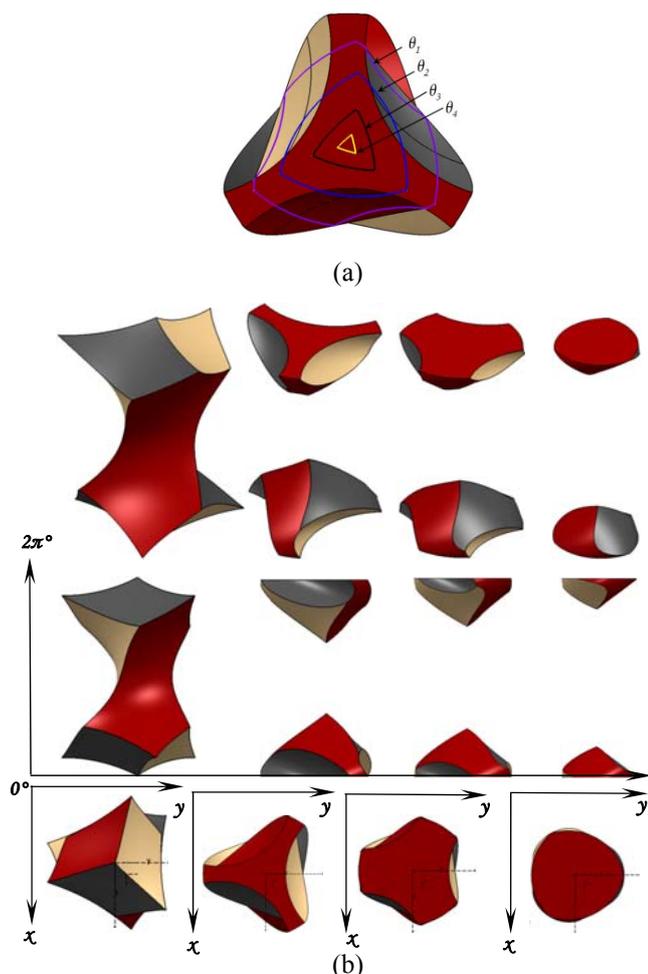


Figure 15 Slices within the workspace (a), workspaces for different plane altitudes (b)

Thus, we can resume this proposed geometrical algorithm used in this paper by the following successive steps:

- Step 1. Set up of the three vertex spaces associated with each limb.
- Step 2. Parameterize a projective plane  $\mathcal{P}$  with two rotation angles  $\alpha$  and  $\beta$  which correspond to the orientation parameters of the moving platform around the x- and y-axis respectively, and one translation distance  $d$ .
- Steps 3. Intersect the parameterized plane  $\mathcal{P}$  with the vertex spaces to obtain the parameterized surfaces associated with the limbs.
- Step 4. Set up of a helical curve whose axis is normal to the projective plane  $\mathcal{P}$ , and its radius is defined by the offsets  $OC_i$  with  $i=1, 2, 3$ . (See Figure 4(b)).
- Step 5. Sweep the three parameterized surfaces along a helical path to obtain three dimensional potential workspaces associated with the limbs considered independently and having the mobile platform as end-effector.
- Step 6. Apply the Boolean intersection operation on the three workspaces, obtained in step 5, to get the common volume corresponding to the three dimensional workspace of 3-CRS parallel manipulator. The approach proposed to characterize and analyze the workspace can be used for solving several types of optimization problems in PRMs: set up the design parameters for a prescribed workspace, set up the robot placement with respect to a given task, set up the maximal reachable workspace for constrained values of the design parameters.

## 5 CONCLUSION

A new modular architecture of PRM with 6-DOFs is presented in this paper. A geometrical approach, implemented in CATIA CAD tool, is used to determine and characterize the workspace. The influence of the joints limits is investigated to determine the actual constant orientation workspace. Finally, a new method to represent and characterize the workspace is proposed. This method can be extended to any spatial PRM with 6-DOFs.

## REFERENCES

- [1] Masory O., Wang J., Workspace evaluation of Stewart platforms, *ASME Robotics, Spatial Mechanisms, and mechanical Systems*, Vol. 45, pp. 337-346, 1992.
- [2] Bonev I.A., Ryu J., A geometrical method for computing the constant-orientation workspace of 6-PRRS parallel manipulators, *Journal of Mechanism and machine theory*, Vol. 36, pp. 1-13, 2001.
- [3] Pusey J., Fattah A., Agrawal S., and Messina E., Design and workspace analysis of a 6-6 cable-suspended parallel robot, *Journal of Mechanism and Machine Theory*, Vol. 39, pp. 761-778, 2004.
- [4] Stan S.D., Mătieș V., and Bălan R., Kinematics analysis, design and optimization of a six degrees-of-

- freedom parallel robot, *IEEE, 6th EUROMECH Nonlinear Dynamics Conference*, Saint Petersburg, RUSSIA, June 30-July 4, 2008.
- [5] Zhao J.S., Chen M., Zhou K., Dong J.X., and Feng Z.J., Workspace of parallel manipulators with symmetric identical kinematic chains, *Journal of Mechanism and machine theory*, Vol. 41, pp. 632-645, June 2006.
- [6] Merlet J.P., *Parallel robots*, 2nd edition, Springer, Dordrecht, 2006.
- [7] Shawn P.A., Robert B.J., and Robert L.D., Comparison of discretization algorithms for NURBS surfaces with application to numerically controlled machining, *Journal of Computer-Aided Design*, Vol. 29, No. 1, pp. 71-83, 1997.
- [8] Kumar A., and Waldron K.J., The workspaces of a mechanical manipulator, *ASME Journal of Mechanical Design*, Vol. 103, pp. 665-672, July, 1981.
- [9] Yang D.C., Lee T.W., On the workspace of mechanical manipulators, *ASME Journal of Mechanisms, Transmission and Automation in Design*, Vol. 105, pp. 62-69, 1983.
- [10] Waldron K.J., Raghavan M., and Roth B., Kinematics of a hybrid series-parallel manipulation system, *ASME Journal of Dynamic Systems, Measurement and Control*, Vol. 111, pp. 211-221, 1989.
- [11] Chedmail P., Dombre E., and Wenger Ph., *The CAD in Robotic, Tools and Methodologies*, HERMES, 1998.
- [12] Wenger Ph., Chablat D., Uniqueness Domains in the Workspace of Parallel Manipulators, *IFAC-SYROCO*, Vol. 2, pp. 431- 436, 3-5 Sept. Nantes, 1997.
- [13] Gosselin C., Determination of the workspace of 6-dof parallel manipulators, *ASME Journal of Mechanical Design*, Vol. 112(3), pp. 331-336, 1990.
- [14] Merlet J.P., Geometrical determination of the workspace of a constrained parallel manipulator. *In ARK*, pp. 326-329, Ferrare, 7-9 September 1992.
- [15] Kohli D., Spanson J., Workspace Analysis of Mechanical Manipulators Using Polynomial Discriminates, *ASME Journal of Mechanisms, Transmissions and Automation in Design*, Vol. 107, pp. 209-215, 1985.
- [16] Gogu G., Mobility and spatiality of parallel robots revisited via theory of linear transformations, *European Journal of Mechanics A/Solids*, Vol. 24, pp. 690-711, 2005.
- [17] Dombre E., *Analyse et modélisation des robots manipulateurs*, Chapitre 5, Hermes Science Publications, Paris, 2001.

# AN EXPERIMENTAL EVALUATION OF EARTHQUAKE EFFECTS ON MECHANISM OPERATION

Stanislav Sula\*      Marco Ceccarelli\*\*      Doina Pisla\*

\* Technical University of Cluj-Napoca, Romania,  
\*\* LARM, University of Cassino, Italy

## ABSTRACT

The results of successful experiments for investigating the earthquake effects on mechanism operation by using Cassino Parallel Manipulator (CaPaMan) as an earthquake simulator are presented. The experimental tests have been carried on by using a slider-crank and a four bar linkage as representing machine operations. The mechanism behavior has been experienced as strongly influenced when the mechanism motion has low speed as compared to earthquake disturbances. But even at high speed mechanisms are affected by the variable earthquake frequency.

Keywords: Experimental Mechanics, Earthquake simulation, Mechanism operation

## 1 INTRODUCTION

A 3 DOF parallel manipulator is presented in this paper with experimental results regarding the earthquake motion and the effects on mechanisms operation, as an extension of previous work on this subject [1]-[4].

Civil constructions and engineering structures are designed to withstand a variety of operational loads and environmental conditions over decades, in safe and economic usage. Earthquakes are part of these environmental conditions. Therefore, motions that are commonly associated with earthquakes are of great interest and they are reproduced by earthquake simulators in laboratories of Civil Engineering Departments, like for example referred in [5]-[11]. Those simulators show operation functionality with feature that can be ascribed to parallel manipulators. Thus, the choice of using a parallel mechanism for simulating earthquake motions is justified by several advantages, namely the ratios of payload capacity / mass is usually high, because the payload on the mobile platform is distributed among the robot legs and it is permitted to use it in an optimal way, mechanism response can be enough stiff, and motion reproduction can be accurate even in acceleration magnitudes. The current earthquake simulators as the above-mentioned ones are

available but most of them do not take into account and operate for a 3D motion of the terrain as due to the earthquake.

In this work the earthquake simulation problem has been solved by a proper analysis and consequent reproduction of an earthquake motion. The aim was to use the experimental results with CaPaMan (Cassino Parallel Manipulator) prototype at LARM in Cassino, to investigate experimentally an earthquake motion characteristic on the mechanism operation. For that we used a slider-crank mechanism and a four-bar linkage as representatives of machine components and their functionality.

## 2 EARTHQUAKE MOTION CHARACTERISTICS

There are three main types of seismic motion due to an earthquake: normal, reverse (thrust) and strike-slip. There are also three main types of seismic waves as shown in Fig. 1, namely P waves (primary waves) that are longitudinal or compression waves, S waves (secondary waves) that are transverse or shear wave, and superficial waves M (named as Love and Rayleigh). In addition, they can be classified by referring both to the propagation speed and terrain movements. The intensity, shape and duration of a seismic motion can depend also on the characteristics of the terrain through which the seismic waves propagate. In Fig. 1c) the main difference among the seismic waves is represented in terms of acceleration magnitude and characteristic period of oscillating motion.

---

Contact author: Stanislav Sula<sup>1</sup>, Doina Pisla<sup>2</sup>,  
Marco Ceccarelli<sup>3</sup>

<sup>1</sup>E-mail: stanislav.sula@tcm.utcluj.ro

<sup>2</sup>E-mail: doina.pisla@mep.utcluj.ro

<sup>3</sup>E-mail: ceccarelli@unicas.it

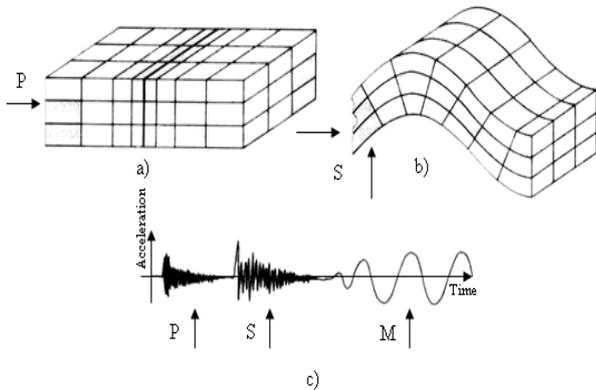
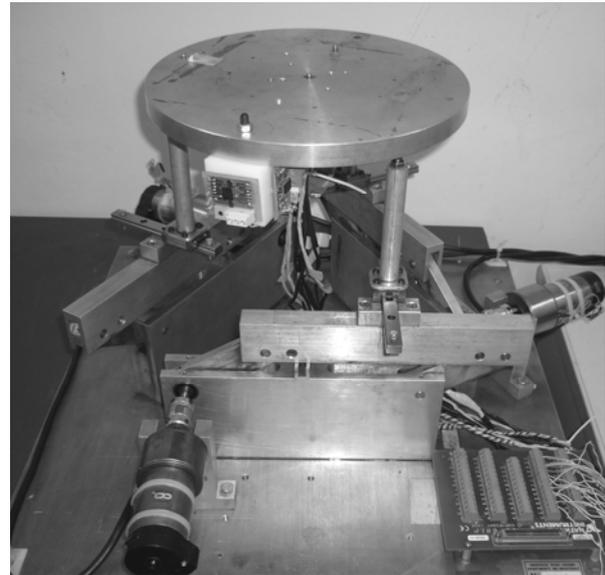


Figure 1 Basic characteristic of seismic waves: a) P waves; b) S waves; c) Types of seismograms.

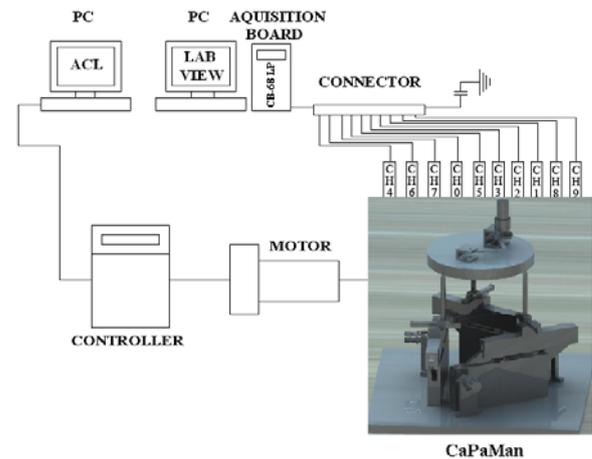
### 3 AN EXPERIMENTAL SETUP

The prototype illustrated in Fig. 2a) is the CaPaMan (Cassino Parallel Manipulator) prototype which has been proposed, designed and built at LARM in Cassino since 1997 [16]. The laboratory test-bed prototype for earthquake simulator consists of the CaPaMan parallel manipulator, an acquisition board connected to the computer in order to acquire linear accelerations occurring along the axes of the reference system belonging to the mobile platform, a controller, and nine accelerometers. Six of them are accelerometers from Kistler Company, and the other three accelerometers (ADXL 105) were settled up in LARM with a low-cost design features. The accelerometers have been properly installed on the faces of three cubes made of plastic and they were installed in points H (Hx, Hy, Hz), P1 (P1y, P1z) and P2 (P2x, P2y, P2z). Another accelerometer has been placed in point P3 (P3z), as shown in Fig. 2c) on the structure that was fixed under the platform.

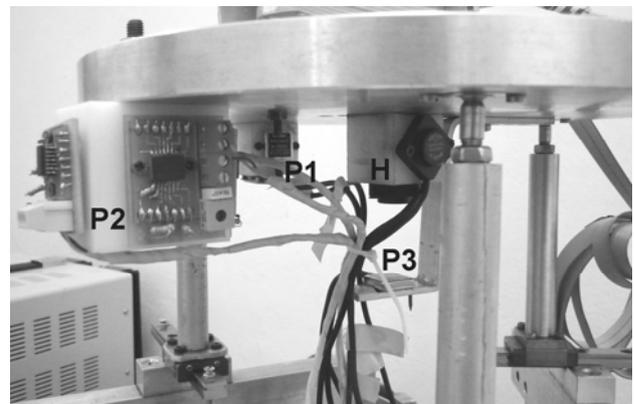
Nine accelerometers were used to monitor the acceleration of the mobile platform during simulation of earthquake motions. The acquisition system for CaPaMan is composed of the accelerometers on the parallel manipulator that are connected to the acquisition board, that sends information signals to the computer through the CB-68 LP parallel port. The information is processed through suitable Virtual instruments in LabView software. The programming and control of the CaPaMan is performed by means of a robot controller Scorbot-ER V. Scorbot-ER V is commercial robot whose controller has been used for its user-oriented characteristics in terms of simple connection to servo controlled actuators, and for ACL programming language. Preliminarily, CAD models for CaPaMan, 4 bar linkage and slider-crank mechanism have been developed for simulation purposes to understand the peculiarities of the CaPaMan applications and its feasibility for m engineering viewpoints. The 3D model of CaPaMan has been developed by Solid Works software, as shown in Fig. 3a).



a)



b)



c)

Figure 2 CaPaMan (Cassino Parallel Manipulator) as earthquake simulator: a) Built prototype; b) Scheme of system layout; c) Location of the nine accelerometers on the mobile platform of CaPaMan.

It consists of the three motors (actuators), the fixed platform, the parallelogram mechanism, the prismatic joints, the connecting bars, the spherical joints and the mobile platform. A 4 bar linkage mechanism has been fixed on top of the mobile platform, and it is designed in order to test its behavior when subjected to seismic motion simulation. Also the slider-crank mechanism, Fig. 3b), has been fixed on the platform of CaPaMan. In this work, a 4-bar linkage mechanism and a slider-crank mechanism have been chosen to be subjected to horizontal, two-directional and 3D motions of the mobile platform of the CaPaMan as simulating several types of earthquake motions. Experimental set up with mechanisms are shown in Fig. 4.

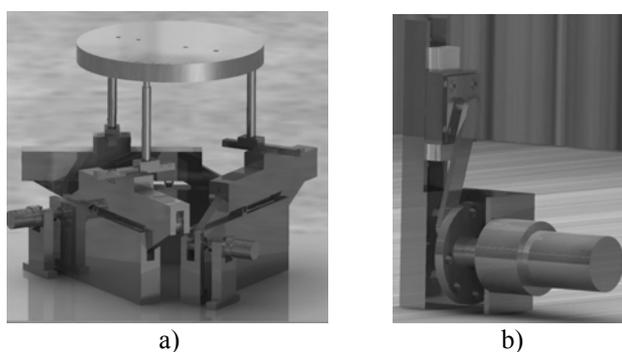


Figure 3 3D CAD models: a) CaPaMan; b) Slider-crank mechanism

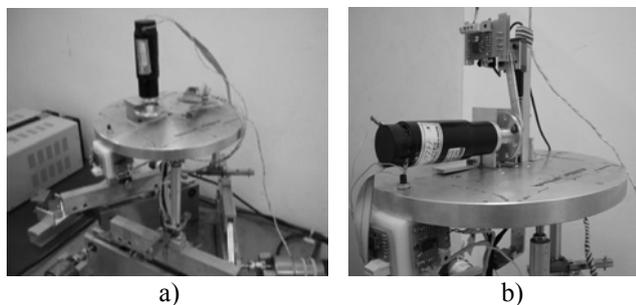


Figure 4 Mechanism installation: a) Slider-crank mechanism; b) 4 bar linkage.

#### 4 RESULTS OF EXPERIMENTAL TESTS

The three types of earthquakes that CaPaMan can reproduce as Horizontal, Two-direction motion and 3D earthquake motion have been tested. Each experiment was repeated three times for statistical purposes in order to have significant results. The input data for the simulated earthquakes have been considered as the type of earthquake, input parameters (angle of oscillations, duration and number of oscillations), maximum peak of the acceleration related to axis Ox and Oz of the mobile platform and for the mechanisms, the angular velocity and acceleration, the experimental date and file name. Also the velocity of the motor that actuate the wheel of the mechanism was set on 30, 60, 120, 150, 300 RPM. In Tab.I series of experiments are reported as related to Horizontal earthquake motion program. The experiments were

conducted by keeping constant both the angle and the time. The number of oscillations was modified from test to test. Specified times are expressed for a half-oscillation of the simulated seismic motion wave. The tests have been developed by modifying the frequency, and it started from frequencies of 1 Hz up to 3 Hz, by changing also the angle of the legs. It can be observed that the maximum accelerations are obtained at Hx and P2x, with constant amplitudes and reaching values of about  $18 \text{ m/s}^2$  at a constant frequency of 1.5 Hz, as outlined in Fig. 5. The medium amplitude of the rest of the accelerations is  $2 \text{ m/s}^2$ , at the same frequency of 1.5 Hz. The experiments using the program Verquake (Two directional earthquake), that are summarized in Tab. I, has been conducted by maintaining constant the time and the oscillation amplitude. Each experiment has been repeated three times in order to have significant results. Judging by the results obtained from experiment no. R2d050509 shown in Fig. 6, it can be observed that from the acquisitions belonging to Hz accelerometer the frequency obtained is maximum 3 Hz, in other words three oscillations per second. It is also possible to observe that the motion of Hz is similar with the one of P3z. The vertical accelerations obtained from the experiments no. R2d050509 have together medium amplitude of  $2.5 \text{ m/s}^2$  with a constant amplitude at Hz and P3z while P1z, P3z and Mz are being influenced by secondary movements belonging to the mobile platform during its vertical translation as known as coupled motion. For the case of three-dimensional seismic wave simulations, the simulated earthquake has replicated a seismic event that occurred in Assisi in 1997. 3Dquake program is able to give to the mobile platform the typical movement of a rigid plane when subjected to a seismic event, by using an appropriate input motion of the actuated shafts. The frequency and the amplitude of the oscillations are the parameters that the programmer introduces in order to obtain the whole domain of frequencies and amplitudes. The undulating movement of the CaPaMan mobile platform is obtained by moving one of the three driving shafts with a certain angle, after that actuating the other two shafts with the same angle. The movement that is obtained is a first translation towards the base, and then another translation of the other extremity of the platform, followed by a repeated translation in the other direction. Simultaneously a rotation is obtained around the Z axis and a translation along the X axis. Regarding to a low speed of the mechanism, with 30 rpm, it can be observed that acceleration of the mechanism related to X axes is almost zero, the maximum peak it is up to  $0.9 \text{ m/s}^2$ . Acceleration related to Mx in stationary position of the mobile platform seems to be due to backlashes in the joints, non perpendicular and non planar positioning of the accelerometer on the mechanism and also the environmental effects. From Fig. 8c) it can be observed that acceleration along Z axes starts from zero and easily decreases to  $2 \text{ m/s}^2$  after 2 seconds it reaches again the zero value. The acceleration peaks are strict related to TDC (Top Dead Centre) and BDC (Bottom Dead Centre).

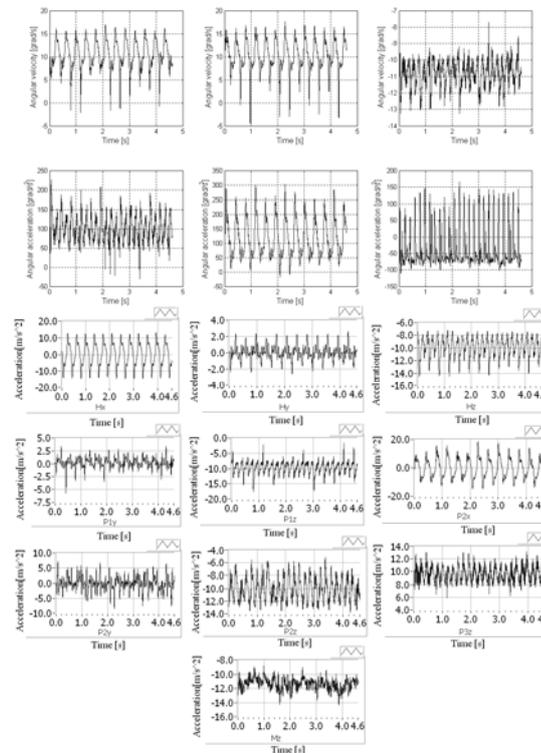


Figure 5 The data acquisitions and the diagrams of the angular speeds ( $\omega_x$ ,  $\omega_y$  and  $\omega_z$ ) in degree/s and accelerations ( $\dot{\omega}_x$ ,  $\dot{\omega}_y$  and  $\dot{\omega}_z$ ) in degree/s<sup>2</sup> with respect to test nr. R1b050509 in Tab. 1

When the slider goes down the acceleration peaks are decreased below zero, and when it goes up it reaches the zero value. When the mechanism is subjected to a 3D earthquake, the acceleration along X axes has a radical change, from 0.9 m/s<sup>2</sup> up to 11 m/s<sup>2</sup>. The difference between the stationary position of the mobile platform and the one subjected to earthquake is 1 g. The acceleration along the Z axes has a change to 8.8 m/s<sup>2</sup>, and the difference between the stationary positions is 0.7 g. To have a more clarity about the effects of earthquake on mechanism operation, the speed of the mechanism was raised from 30 rpm to 60, 120, 150 and 300 rpm. The seismograms related to this speed are represented in Fig.10 and Fig.11. The experimental results have shown that the effect of earthquake is inverse proportional with the speed of the mechanism. A final conclusion can be added as that the slider-crank mechanism is really affected by the earthquake when it has a low speed, when it reaches a speed closed to the frequency of the earthquake it will also be affected but not so insignificantly as comparison with the case with a smaller speed. The second study of earthquake effects on mechanism was carried out on a 4 bar-linkage mechanism that was subject only to 3D

earthquake. The experiments were conducted at different speed of the mechanism 30, 60, 120, 150, 300 rpm (0.5, 1, 2, 2.5, 5 Hz) and also different frequencies of the earthquake (1, 1.5, 2.5 Hz). Each experiment was repeated for three times to make a statistical comparison and have significant results. The four bar linkage mechanism is actuated by a rotary motor, which transmits the movement to a driving wheel, which at its turn moves the 2 bars. The results are almost the same as for the case with the slider-crank mechanism. For a better understanding of the operation behavior of the 4 bar-linkage a sensor was installed on the coupler of the mechanism. In Fig. 9a), a zoomed view of the seismogram is shown as related to stationary position of the CaPaMan. A slightly displacements can be observed along the time. That means that acceleration of the 4 bar linkage mechanism is very small up to 1.4 m/s<sup>2</sup> at the speed of 30 rpm. Subjected to 3D earthquake the acceleration suddenly rises up to 16.7 m/s<sup>2</sup>.

Table I - Data and results for test of linear simulated earthquake with slider-crank mechanism operating at 30 rpm. \*values for the stationary position of the platform

Description	Input parameters			Results(m/s <sup>2</sup> )			Data file name
	$\Delta\alpha$ (°)	T (s)	Hz	$a_{Hmax}$ H.	$a_{Hmax}$ V.	$a_{mech}$ V. *1.9	
Horquake with slider-crank mech.	20	0.1	1	14.2	4.6	3.9	R1a050509
	20	0.1	1	17.1	4.5	4.4	R1a110509
	20	0.1	1	16.9	5.3	6.9	R1a120509
	20	0.1	1.5	14.7	4.7	4.5	R1b050509
	20	0.2	1.5	13.9	3.6	3.4	R1d050509
	20	0.2	1.5	12.2	4	4.9	R1d110509
	20	0.2	1.5	13.9	4.9	6.3	R1d120509
	30	0.2	1.5	19.2	8.1	8.3	R1e050509
	30	0.2	1.5	19.2	8.9	9.6	R1e110509
	30	0.2	1.5	19.8	8	9.1	R1e120509
Two-Direction. Earthquake with slider-crank mech.	20	0.1	1	2.5	2.3	2.8	R2a050509
	20	0.1	1	2.9	2.3	3.7	R2a110509
	20	0.1	1	3.9	2.6	4.1	R2a120509
	30	0.1	3	2.4	5.9	5.2	R2d050509
	30	0.1	3	4.9	5.6	5	R2d110509
	30	0.1	3	3.8	5.3	5	R2d120509
	30	0.2	1.5	1.4	3.5	2.8	R2e050509
	30	0.2	1.5	1.7	3.8	2.9	R2e110509
30	0.2	1.5	1.6	3.4	3.4	R2e120509	

Table II - Data and results for tests of 3D simulated earthquake with slider-crank mechanism operating at 30 rpm. \* values for the stationary position of the platform

$\Delta\alpha$ (°)	T (s)	Hz	$a_{mech}$ V *2	$a_{mech}$ H *0.9	$\omega_{max}$ [°/s]	$\dot{\omega}_{max}$ [°/s <sup>2</sup> ]	Data file
30	40	1	7.2	11.2	20.6	314.3	R3a180509
30	40	1	8.8	11	24.6	384.9	R3a200509
30	40	1	9.2	10.5	24.2	375.1	R3a220509
30	50	1.5	7.6	11	21.1	305.7	R3b180509
30	50	1.5	7.9	10.6	24.6	371.6	R3b200509
30	50	1.5	9.5	10.7	25.2	369.8	R3b220509
30	60	2.5	10.9	11	20.9	326.3	R3c180509
30	60	2.5	8.9	11.2	24.9	382.8	R3c200509
30	60	2.5	8.9	11.1	25	360.2	R3c220509

Table III - Data and results for tests of 3D simulated earthquake with slider-crank mechanism operating at 150 rpm. \* values for the stationary position of the platform

$\Delta\alpha$ (°)	T (s)	Hz	$a_{mech}$ V. m/s <sup>2</sup> *9.7	$a_{mech}$ H m/s <sup>2</sup> *3.7	$\omega_{max}$ [°/s]	$\dot{\omega}_{max}$ [°/s <sup>2</sup> ]	Data file
30	40	1	13.3	10.9	20.6	314.3	R3a180509
30	40	1	11.5	11.4	24.6	384.9	R3a200509
30	40	1	13.6	10.9	24.2	375.1	R3a220509
30	50	1.5	12	11.2	21.1	305.7	R3b180509
30	50	1.5	10.7	11.4	24.6	371.6	R3b200509
30	50	1.5	12.2	11.4	25.2	369.8	R3b220509
30	60	2.5	13	11.6	20.9	326.3	R3c180509
30	60	2.5	11.2	10.9	24.9	382.8	R3c200509
30	60	2.5	12.7	10.9	25	360.2	R3c220509

Table IV - Data and results for tests of 3D simulated earthquake with slider-crank mechanism operating at 300 rpm. \* values for the stationary position of the platform

$\Delta\alpha$ (°)	T (s)	Hz	$a_{mech}$ V *20.7	$a_{mech}$ H *5.4	$\omega_{max}$ [°/s]	$\dot{\omega}_{max}$ [°/s <sup>2</sup> ]	Data file
30	40	1	24.1	12.2	20.6	314.3	R3a180509
30	40	1	25.5	11.1	24.6	384.9	R3a200509
30	40	1	24.9	12.1	24.2	375.1	R3a220509
30	50	1.5	25.7	10.6	21.1	305.7	R3b180509
30	50	1.5	25.2	12.4	24.6	371.6	R3b200509
30	50	1.5	24.8	11.7	25.23	369.8	R3b220509
30	60	2.5	24.8	11.7	20.9	326.3	R3c180509
30	60	2.5	24.6	11.9	24.9	382.8	R3c200509
30	60	2.5	25	11.4	25	360.2	R3c220509

This means a difference between the stationary position of the mobile platform and the one subjected to earthquake of 1.5 g in other words the disturbance is 15 times bigger than the one from stationary position. To have a better understating we have increased the speed gradually and the effects are the same as we had with the slider-crank mechanism, that the effect of earthquake is inverse proportional with the speed of the 4 bar-linkage. The seismograms related to this speed are represented from Fig.12 and Fig.13.

Table V - Data and results for tests of 3D simulated earthquake with 4 bar-linkage mechanism operating at 300 rpm. \* values for the stationary position of the platform

$\Delta\alpha$ (°)	T (s)	Hz	$a_{mech}$ H	$\omega_{max}$ [°/s]	$\dot{\omega}_{max}$ [°/s <sup>2</sup> ]	Data file
30	40	1	21.7	20.6	314.3	R3a180509
30	40	1	23.6	24.6	384.9	R3a200509
30	40	1	23.3	24.2	375.1	R3a220509
30	50	1.5	21.5	21.1	305.7	R3b180509
30	50	1.5	22.6	24.6	371.6	R3b200509
30	50	1.5	22.8	25.2	369.8	R3b220509
30	60	2.5	20.8	20.9	326.3	R3c180509
30	60	2.5	23	24.9	382.8	R3c200509
30	60	2.5	24.3	25	360.2	R3c220509

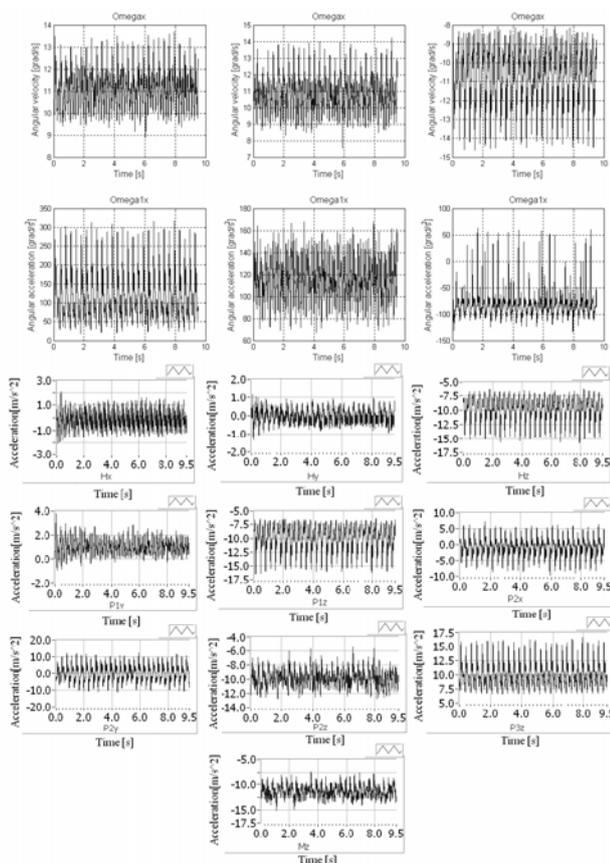


Figure 6 The data acquisitions and the diagrams of the angular speeds ( $\omega_x$ ,  $\omega_y$  and  $\omega_z$ ) in degree/s and accelerations ( $\dot{\omega}_x$ ,  $\dot{\omega}_y$  and  $\dot{\omega}_z$ ) in degree/s<sup>2</sup> with respect to test no. R2d050509 in Tab. I

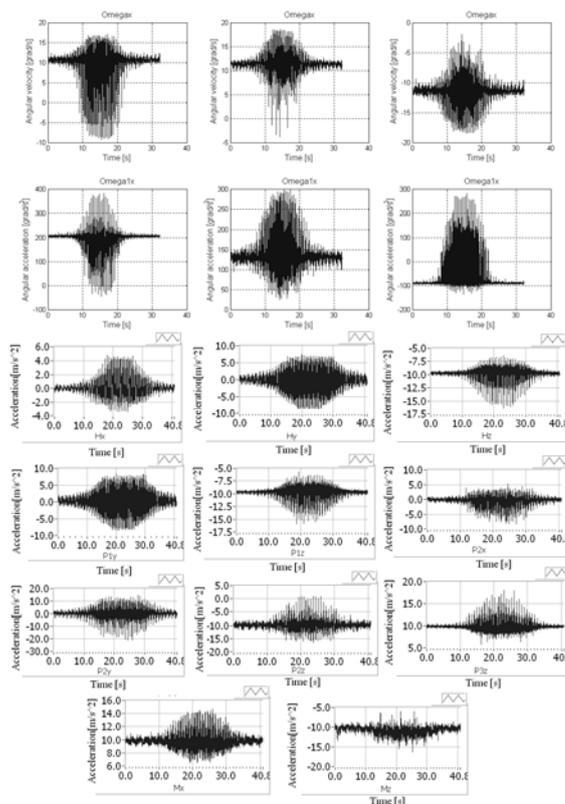


Figure 7 The data acquisitions and the diagrams of the angular speeds ( $\omega_x$ ,  $\omega_y$  and  $\omega_z$ ) in degree/s and accelerations ( $\dot{\omega}_x$ ,  $\dot{\omega}_y$  and  $\dot{\omega}_z$ ) in degree/s<sup>2</sup> with respect to test no. R3c180509 in Tab. II

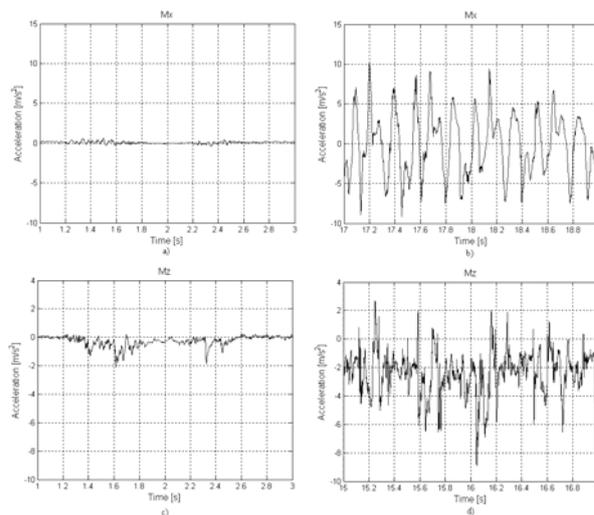


Figure 8 Zoomed view of results of test no. R3c200509 from Tab.II for slider-crank mechanism moving at 30 rpm as measured for Mx and Mz when is subjected to: a), c) Stationary position; b), d) 3D earthquake.

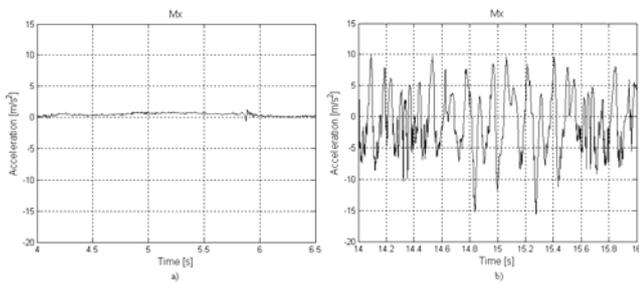


Figure 9 Zoomed view of results of test no. R3c200509 for 4 bar linkage mechanism moving at 30 rpm as measured for Mx when is subjected to: a) Stationary position; b) 3D earthquake.

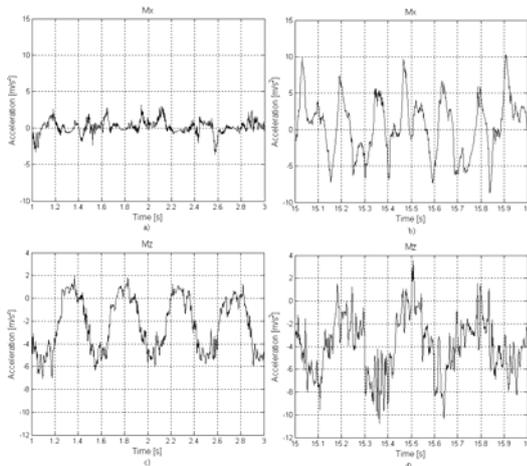


Figure 10 Zoomed view of results for tests no. R3c200509 from Tab.III for slider-crank mechanism operating at 150 rpm as measured for Mx and Mz when is subjected to: a), c) Stationary position; b), d) 3D earthquake.

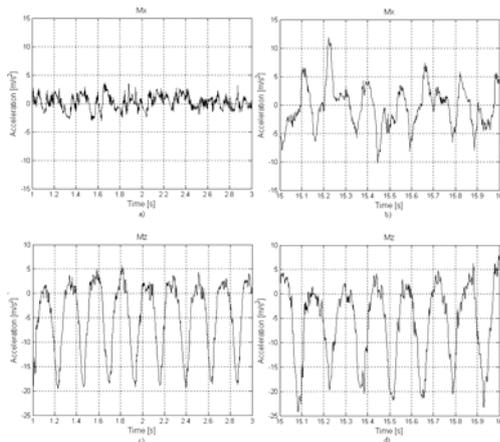


Figure 11 Zoomed view of results for test no. R3c200509 from Tab.IV for slider-crank mechanism operating at 300 rpm as measured for Mx and Mz when is subjected to: a), c) Stationary position; b), d) 3D earthquake.

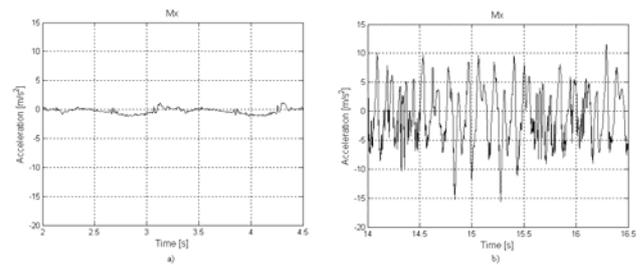


Figure 12 Zoomed view of results for test no. R3c200509 for 4 bar linkage mechanism operating at 60 rpm as measured for Mx: a) Stationary position; b) 3D earthquake.

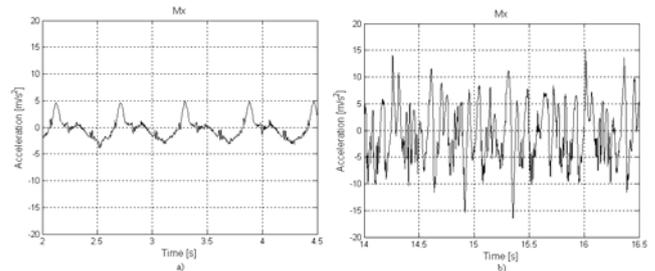


Figure 13 Zoomed view of results for test no. R3c200509 for 4 bar linkage mechanism operating at 120 rpm as measured for Mx: a) Stationary position; b) 3D earthquake

5 CONCLUSIONS

In this paper a slider-crank mechanism and a four bar linkage have been chosen to simulate complex motions of machinery by looking at the motion of the coupler and how it can be affected by seismic disturbance. It has been experienced through these experiments that mechanisms can be affected by earthquake disturbances when they are operated at low speed as compared with the frequency of an earthquake. But even when mechanisms are operated at high speed a significant influence can be experienced as function of the earthquake frequency and magnitude. Moreover we presented an experimental layout and its feasibility for an experimental characterization of mechanism operation under disturbance due to earthquake motion which is quite different from a usual external disturbance due to its short duration and large variability in motion characteristics. On the other hand a theoretical study and a suitable modelling can be useful to a better use of the validation of results.

REFERENCES

[1] Ceccarelli M., Ottaviano E., Galvagno M., A 3-DOF Parallel Manipulator as Earthquake Motion Simulator, *Proceedings of the 7th International Conference on Control, Automation, Robotics and Vision ICARCV 2002*. Singapore, 2002, pp. 944-949.  
 [2] Carvalho J.; Ceccarelli M., Seismic motion simulation based on Cassino Parallel Manipulator, *Journal of the*

- Brazilian Society of Mechanical Sciences*. Vol.25, n.3, July 2002, pp.213-219.
- [3] Ottaviano E., Ceccarelli M., Castelli G., Experimental Results of a 3-DOF Parallel Manipulator a an Earthquake Motion Simulator. *ASMEDETC, Mechanisms and Robotics Conference*. Salt Lake City, 2004, paper DETC2004-57075.
- [4] Ottaviano E., Ceccarelli M., An Application of a 3-DOF Parallel Manipulator for Earthquake Simulations. *IEEE Transactions on Mechatronics*. Vol. 11, No. 2, pp. 240-246, 2006.
- [5] ISMES, *The Six Degrees of Freedom Shaking Table, MASTER*. Enel Hydro B.U. ISMES, Seriate, 2001.
- [6] Structural Dynamics Control & Earthquake Engineering Laboratory, *Earthquake Simulator Facilities*. Department of Civil Engineering and Geological Sciences, University of Notre Dame, Notre Dame, <http://www.nd.edu/~quake/facilities/>, 2003.
- [7] Structural Engineering Group, *Tri-axial Earthquake and Shock Simulator (TESS)*. U.S.Army Construction Engineering Research Laboratories, [www.cecel.army.mil/fl/seg/tess.htm](http://www.cecel.army.mil/fl/seg/tess.htm), 2003.
- [8] VCE (Vienna Consulting Engineers), *Shaking Table Test at ISMES Institute*. [www.vce.at/pages/int\\_indhtm?/pages/inter/ital.htm](http://www.vce.at/pages/int_indhtm?/pages/inter/ital.htm), 2003.
- [9] Washington University, *Laboratory Facilities*. Washington, <http://wusceel.civewustl.edu/quake/facilities.htm>
- [10] National Information Service for Earthquake Engineering, *Earthquake Simulator Laboratory*. University of California, Berkeley, <http://peer.berkeley.edu/lab/>, 2003.
- [11] Nevada seismological Lab, *Living with Earthquakes in Nevada: A Nevadan's guide to preparing for, surviving, and recovering from an earthquake*.
- [12] Matthew. O Jackson, *Mechanism Theory, Humanities and Social Sciences*. p 228-77, California Institute of Technology Pasadena, 2003.
- [13] Hunt. K. H., *"Kinematic Geometry of Mechanisms"*. Oxford University Press, New York, 1978.
- [14] Ceccarelli M., A New 3 D.O.F. Parallel Spatial Mechanism. *Mechanism and Machine Theory*, Vol. 32, No. 8, pp. 895-902, 1997.

# EXERCISE DEVICE FOR UPPER-EXTREMITY SENSORY-MOTOR CAPABILITY AUGMENTATION BASED ON MAGNETO-RHEOLOGICAL FLUID ACTUATOR

Roman Kamnik, Jernej Perdan, Tadej Bajd, Marko Munih

Laboratory of Robotics and Biomedical Engineering

Faculty of Electrical Engineering, University of Ljubljana

Tržaška 25, 1000 Ljubljana, Slovenia

E-mail: {roman.kamnik, jernej.perdan, tadej.bajd, marko.munih}@robo.fe.uni-lj.si

## ABSTRACT

Resistance exercise has been widely reported to have positive rehabilitation effects for patients with neuromuscular and orthopaedic disorders. This paper presents the design of a versatile rehabilitation device in the form of a rotation joint mounted on the adjustable arm support that provides a controlled passive resistance during strength training of hand muscles. The resistance is supplied by a rotational magnetorheological actuator controlled with regards to the force feedback information. The device provides both isometric and isokinetic strength training and is reconfigurable for different usage conditions. The experimental evaluation results show that the usage of magnetorheological actuator is advantageous to the electrical motors in the cases of passive resistance based exercise.

Keywords: rehabilitation, exercise device, magnetorheological fluid

## 1 INTRODUCTION

In the rehabilitation and sports medicine computerized active exercise devices have been shown to be suitable for providing the clinical delivery of training of the required intensity [1]. Especially challenging aspect is the recovery of hand function. We have recently developed a novel system for hand sensory-motor augmentation [2] which is designed to allow force tracking training of finger flexors and extensors and to provide objective data on training performance in isometric conditions. Incorporated functional electrical stimulation adds to reduced finger force generation due to injury, thus motivating the user for better achievements. The system consists of a visual feedback display, the hand force measuring device, and the closed-loop controlled electrical stimulator. The results of pilot therapy study in incomplete tetraplegic subjects showed that augmentation of voluntary grip force control with presented system is possible.

However, the training performed in isometric conditions in which at various angular positions the external resistance applied to the joint is always equal to the force applied by the patient is not considered as most efficient.

As more efficient, the isotonic and isokinetic exercises are considered. The isotonic exercise is performed dynamically over a predefined range of motion. The resistance applied to the joint is either constant or follows a predefined pattern as function of joint angular position. This mode of exercise is motivated by the length tension relationship of skeletal muscle in which largest force is generated when muscle fibers are at their optimal length. The force producing capacity of a muscle changes across the range of joint motion and is typically highest in the midrange of joint motion. Muscle force generation during concentric exercise is also influenced by the contraction velocity as described by a hyperbolic model relating the force and velocity during contractions. As contraction velocity increases, muscle force decreases. From this relationship the isokinetic exercise is motivated which is also performed dynamically but in that case, the resistance is applied to the joint only if a predefined angular speed is reached by the joint in order to avoid that the joint exceed this speed value. This particular exercise mode is the only one that enables

---

Contact author: Roman Kamnik

Faculty of Electrical Engineering, University of Ljubljana  
Tržaška 25, 1000 Ljubljana, Slovenia

dynamic training at the maximum muscle force over the entire range of motion.

Most force feedback devices that are capable of regulating joint motion according to the needs of particular patient and take muscle and limb dynamics into consideration rely on electric motors, pneumatics, or some other conventional power producing method.

In this paper we present a semi-active exercise system based on magnetorheological fluid (MR fluid) actuator [3]. This semiactive controlled device can be considered as one that has properties that can be adjusted in real time but cannot input energy into the system being controlled. Such devices typically have very low power requirements and offer the reliability of passive devices, while maintain the versatility and adaptability of fully active systems [4], [5]. In the second section of the paper the principle of operation of MR fluid actuator is presented. The third section presents the design of exercise device based on MR fluid actuator, while the fourth section outlines the experimental results.

## 2 PRINCIPLE OF OPERATION OF MR FLUID ACTUATOR

MR fluids are materials that respond to an applied magnetic field with a change in rheological behaviour. Typically, this change is manifested by the development of a yield stress that monotonically increases with applied field. The MR fluid typically consists of micron-sized, magnetically polarizable particles dispersed in a carrier medium such as mineral or silicone oil. When a magnetic field is applied to the fluids, particle chains form, and the fluid becomes a semi-solid and exhibits viscoplastic behaviour. The MR fluid can be readily controlled with a low voltage (e.g., 12-24 V), current-driven power supply outputting only 1-2 A. The behaviour of MR fluids is often described as a Bingham plastic model having a variable yield strength, which depends upon the magnetic field  $B$ . At fluid stresses below the yield stress the fluid acts as a viscoelastic material exhibiting Newtonian-like behavior. At fluid shear stresses above the field-dependent yield stress the fluid flow is governed by the Bingham plastic equation [9]. This behaviour is described by (1):

$$\tau = \begin{cases} G\dot{\gamma} & \tau < \tau_{yd} \\ \tau_{yd}(B) + \mu_p \dot{\gamma} & \tau > \tau_{yd} \end{cases} \quad (1)$$

where  $B$  is the magnetic field,  $\dot{\gamma}$  is the fluid shear rate and  $\mu_p$  is the plastic viscosity (i.e., viscosity at  $B = 0$ ),  $G$  is the complex material modulus (which is also field dependent).  $\tau_{yd}$  in equation (1) is a function of the magnetic field  $B$  and exponentially increases with respect to magnetic flux density. The relationship is given by:

$$\tau_{yd}(B) = \kappa B^\beta \quad (2)$$

where proportional coefficient  $\kappa$  and the exponent  $\beta$  are intrinsic values of the MR fluid, which are functions of various factors such as magnetic field, particle size, particle shape and concentration, carrier fluid, temperature and magnetic saturation. The applied magnetic field  $B$  is

produced within the actuator when current  $i$  is supplied to the electromagnet encircling the MR fluid, i.e.,

$$B = \kappa_r i \quad (3)$$

True MR fluid behaviour exhibits some significant departures from this simple model. Perhaps the most significant of these departures involves the non-Newtonian behaviour of MR fluids in the absence of a magnetic field.

In general, the MR devices involve either disc-type or valve-type designs. In valve type designs, the fluid is pushed through a narrow channel where the magnetic field is applied to control the flow rate, and hence the applied force. Typically, these designs resemble a cylinder-piston assembly with the coil on the piston haft. In disc-type designs, the fluid is in a narrow gap between a rotating disc and a fixed outer casing [6], [7]. The coil is positioned close to the outer edge of the disc. When the magnetic field is applied, the increased yield stress of the fluid creates a braking torque on the disc.

The braking torque  $T_b$  developed by the MR fluids in the disc-type actuator can be determined as:

$$T_b = 2\pi \int_{r_w}^{r_z} \tau r^2 dr = 2\pi \int_{r_w}^{r_z} (\tau_{yd} + \mu_p \dot{\gamma}) r^2 dr \quad (4)$$

where  $r_z$  and  $r_w$  are the outer and inner radii of the actuator disk, respectively; and  $\dot{\gamma} = r\omega/h$  where  $\omega$  is the angular velocity of the rotating disk and  $h$  is the thickness of the MR fluid gap [10]. Following (2), the equation (4) can be rewritten as:

$$T_b = 2\pi \int_{r_w}^{r_z} (\mu_p \frac{r\omega}{h} + \kappa B^\beta) r^2 dr \quad (5)$$

Integrating (5) and substituting with (3) the braking torque developed by MR fluids can be calculated:

$$T_b = \frac{2\pi}{3} \kappa \kappa_r^\beta (r_z^3 - r_w^3) i^\beta + \frac{\pi}{2h} \mu_p (r_z^4 - r_w^4) \omega \quad (6)$$

Equation (6) shows that the braking torque developed in the circular plate MR fluid actuator can be divided into a magnetic field dependent induced yield stress component  $T_B$  and a viscous component  $T_\mu$ :

$$T_b = T_B + T_\mu = k_i i^\beta + k_\omega \omega \quad (7)$$

## 3 MR FLUID ACTUATOR EXPERIMENTS

For actuating the exercise device, a rotary MR fluid actuator produced by Lord Corporation, USA was used [8]. The Lord TFD Device RD-8043-1 is capable to produce up to 12 Nm of axial torque while it is driven by current-driven power supply with the current capabilities of

up to 1.5 A. The device has a position feedback sensor integrated which outputs a PWM signal with duty cycle varying between 5–95% according to the axis position.

The torque output was measured using a test setup with a load cell, a lever arm, and a data acquisition system. The braking torque experiment started with measuring the static torque threshold while manually rotating the actuator axis, first in the clockwise and then in the anti-clockwise direction. The threshold torque which is actually the sum of a static friction and a magnetic field dependent induced yield stress component  $T_B$  was assessed in several repetitions with different input voltages. The graph on Figure 1 presents the absolute values of acquired threshold torque  $T_T$  with regards to the input voltage  $V_C$ . From the results a nonlinear relationship can be observed.

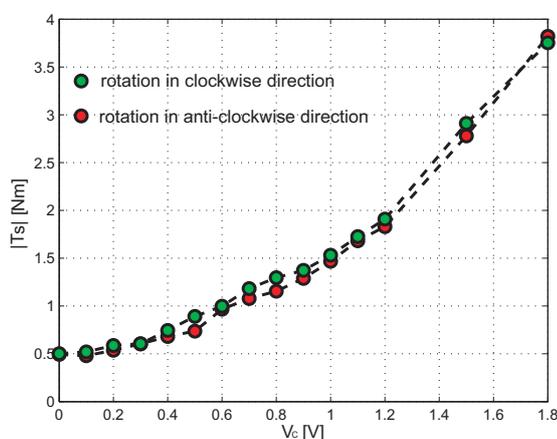


Figure 1 Static torque threshold values with accordance to the MR fluid excitation.

In the second experiment, a dynamic characteristics of MR fluid actuator was measured evaluating the dependence to the motion speed. The braking torque was assessed during motion in forward and backward direction moving with a different rotation velocity and with constant MR fluid actuator input. A family of curves was obtained that is presented in Figure 2. Each curve represents a typical characteristics of the braking torque. The presented values sum the yield stress component  $T_B$ , the viscous component  $T_{\mu}$ , and the static friction. From acquired results a nonlinear torque-velocity relationship with a hysteresis loop can be observed [11].

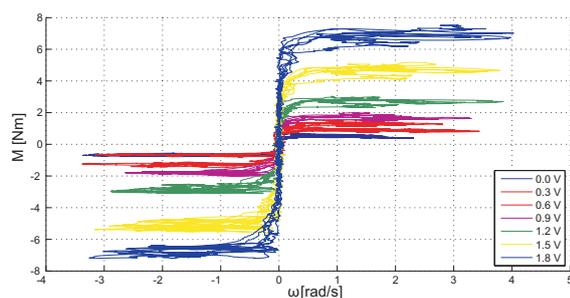


Figure 2 Dynamic characteristics of MR fluid actuator.

#### 4 EXERCISE DEVICE FOR UPPER-EXTREMITY SENSORY-MOTOR CAPABILITY AUGMENTATION

The conceptual scheme of the training system for upper extremity sensory-motor augmentation is presented in Figure 3. The system is designed to train finger or wrist flexor and extensor muscles by performing the position tracking task. The reference and actual positions are displayed on a visual display as two rotational pointers. The MR fluid actuator is used as a braking torque modulating device which allows exercise under isometric, isotonic or isokinetic conditions. The core of the system is a personal computer (PC) that is used for reference generation, actual hand force acquisition, visual presentation of the reference and actual position, and control of the MR fluid actuator. The software application for controlling the system was developed in the Mathworks Matlab-Simulink programming environment and it runs in xPC real time operating system.

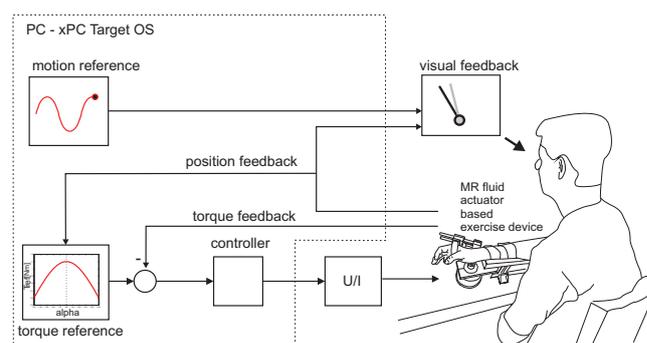


Figure 3 Conceptual scheme of the training system showing its main components: exercise device with force sensor and MR fluid actuator, visual feedback, and closed-loop controller.

The close view of the exercise device is shown in Figures 4 and 5. The device construction is made of aluminium strut elements. On a construction, the MR fluid actuator is mounted, and on its axis an adjustable lever arm with a JR3 force/torque sensors (50M31A-I25; JR3, Inc., Woodland, CA, USA) and a finger fixation are fixed. The fingers are fixed to the force sensor by means of a finger support and a Velcro strap. The finger fixation and force sensor enable the acquisition of the hand forces. To ensure the proper position and to prevent the arm from moving during training, the forearm is fixed to the arm support by Velcro straps. The position of force sensor, the actuator and forearm support is adjustable, allowing the accommodation of measuring setup to each individual, as well as to assess either the right or the left hand. Two PCI boards were used for data acquisition from the force and position sensors, and to generate the control voltage for MR fluid actuator.

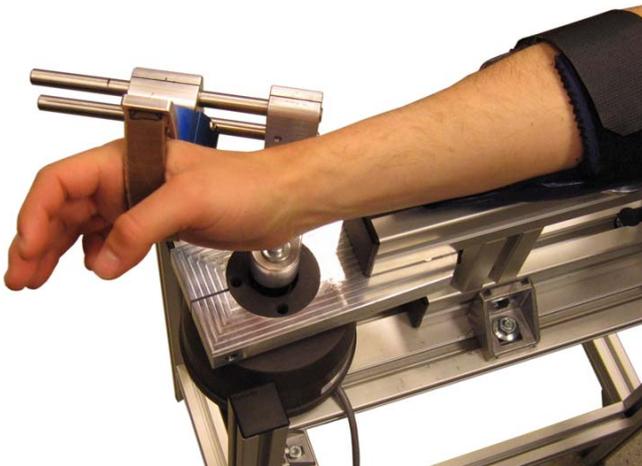


Figure 4 Close view of the exercise device from the left side.



Figure 5 Close view of the exercise device from the right side

## 5 EXPERIMENTAL EVALUATION

To demonstrate the performances of the developed exercise device an experimental evaluation was accomplished. In the position tracking experiment a healthy subject was asked to follow the reference position which was altered linearly in a range of  $\pm 30^\circ$  from its center (fingers extended) position at  $180^\circ$ . During motion, the braking torque was modulated by the MR fluid actuator according to the term:

$$T_{ref} = \pm 0.6 \pm 0.5 * \sin(1.8 * (\alpha - 2.269)) \quad (8)$$

in which parameter  $\alpha$  states for the current position of the actuator axis in radians, and the sign  $\pm$  is changed regarding the rotation direction (clockwise/anticlockwise). According to the term above, the braking torque had the highest value at the finger extended position, while it diminished with displacing from it.

The actual torque was measured while the MR fluid actuator activity was controlled by a PI controller with a feedforward term according to the difference between actual and reference value. Figure 6 presents the actual motion trajectory accomplished during experimental evaluation. In Figure 7 the reference  $T_{ref}$  and actual  $T_{act}$  torques are shown.

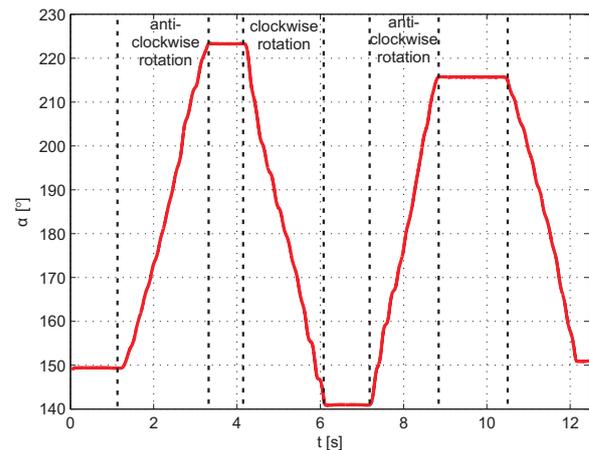


Figure 6 Motion trajectory in experimental evaluation.

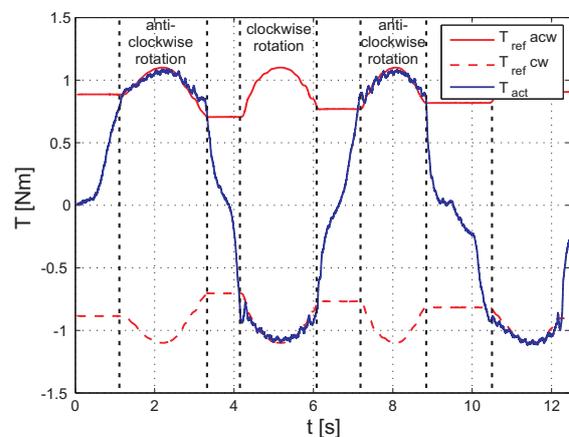


Figure 7 Reference torque tracking of MR fluid actuator.

## 6 CONCLUSION

The paper presents the development and experimental evaluation of semi-active exercise device for upper-extremity sensory-motor capability augmentation. The device is built on the basis of the rotational magnetorheological actuator which allows resistive torque modulation. The frame of the device is constructed to allow flexible change of configuration, while the controller is implemented on a Mathworks Matlab/Simulink environment and real time xPC Target operating system. The experimental results show that the MR fluid actuator is suitable for application in exercise devices as a semi-active element providing braking torque modulation. On its basis,

several exercise modes can be achieved. In comparison to electric motor actuators the power to weight ratio and need for power amplifier is advantageous in the case of MR fluid actuator usage. On the other side, the control is more complex since the MR fluid actuator is a highly nonlinear device.

The proposed areas of application for exercise devices based on the MR fluid actuators are in rehabilitation and sports training.

#### ACKNOWLEDGEMENT

The authors would like to acknowledge the financial support from the Slovenian Research Agency (P2-0228), and Slovenian Technology Agency (3211-05000550).

#### REFERENCES

- [1] Krebs H.I., Hogan N., Aisen M.L. and Volpe B.T., Robot-aided neurorehabilitation. *IEEE Trans. Rehab. Eng.*, Vol. 6, No. 1, pp. 75-87, 1998.
- [2] Perdan J., Kamnik R., Obreza P., Kurillo G., Bajd T. and Munih M., Design and evaluation of a functional electrical stimulation system for hand sensorimotor augmentation, *Neuromodulation*, Vol. 11, No. 3, pp. 208-215, 2008.
- [3] Senkal D. and Gurocak H., Spherical brake with MR fluid as multi degree of freedom actuator for haptics. *J. Intel. Mat. Syst. and Struct.*, Vol. 20, No. 18, pp. 2149-2160, 2009.
- [4] Dong S., Lu K-Q, Sun J.Q. and Rudolph K., A prototype rehabilitation device with variable resistance and joint motion control, *Med. Eng. Phys.*, Vol. 28, No. 4, pp. 348-355, 2006.
- [5] Blake J. and Gurocak H.B., Haptic glove with MR brakes for virtual reality. *IEEE/ASME Tran. Mechatronics*, Vol. 14, No. 5, 2009.
- [6] Li W.H. and Du H., Design and experimental evaluation of magnetorheological brake, *Int. J. Adv. Manuf. Technol.*, Vol. 21, pp. 508-515, 2003.
- [7] Yan Y., Shi-Guo H. and Bo-Seon K., Research on circular plate MR fluids brake, *J. Cent. South Univ. Technol.*, Vol. 14, suppl. 1, pp. 257-259, 2007.
- [8] Jolly M.R., Bender J.W. and Carls J.D., Properties and applications of commercial magnetorheological fluids, *Proc. SPIE - Smart Structures and Materials, Passive Damping and Isolation*, Vol. 3327, pp. 262-275, 1998.
- [9] Guglielmino E., Sireteanu T., Stammers C.W., Ghita G. and Giuclea M., *Semi-active suspension control*, London, England: Springer, 2008.
- [10] Park E.J., Stoikov D., Falcao da Luz L. and Suleman A., A performance evaluation of an automotive magnetorheological brake design with a sliding mode controller, *Mechatronic*, Vol. 16, pp. 405-416, 2006.
- [11] Guo S., Yang S. and Pan C., Dynamic modeling of magnetorheological damper behaviors. *J. Intel. Mat. Syst. and Struct.*, Vol. 17, No. 1, pp. 3-14, 2006.



# LABORATORY TEST FOR AN INTEGRATED DEVICE IN AGRICULTURAL APPLICATIONS

A. Manuello Bertetto \*    C. Falchi \*    R. Pinna \*    R. Ricciu \*\*

\* Dept. of Mechanical Engineering University of Cagliari, Cagliari, Sardinia, Italy

\*\* Dept. of Architecture University of Cagliari, Cagliari, Sardinia, Italy

## ABSTRACT

This paper describes the operating principle of a mechanical system designed to harvest *Crocus Sativus* (saffron) flowers. The former device imitates one of the procedures followed by the pickers, but with the peculiarity that allows to harvest the flower without separating it from its leaves, which is a significant advantage since it simplifies the mechanical detachment of the flower. The system is conceived as a shoulder portable device with two main parts: the first one is specifically designed to detach the flower containing three stigmas, which are the costly final product; the second one is aimed to collect the detached flower through a vacuum collector. The system is carried by the human operator on his back. The paper also deals with some experimental tests in the laboratory carried out in order to highlight the dynamic behaviour of the detaching and of the harvesting devices.

Keywords: mechanical harvesting; agricultural robotics; portable automated device.

## 1 INTRODUCTION

In agricultural harvesting many researchers are engaged to design robotic devices for efficient and delicate agricultural product picking and handling. The level of mechanization in the agricultural field, as well as the choices made and the consequent appropriate use of the tools and techniques developed, has had a direct and considerable effect on the agricultural production rates, on the farms growth and on the environment as well.

Agricultural mechanization has started developing and diffusing since the second half of the twentieth century, when machines for performing either agricultural or zoo technical tasks with well defined operational targets and large economical and social consequences were first realised.

A very high number of devices have been introduced in the agricultural field in order to support or even take place of the work of man, specially when it comes to hard work to be done in open field. In particular many systems and devices are used for products harvesting [1-5].

There is a growing interest in the use of pneumatic devices and robots in horticultural industries. Mechanization of agricultural products harvesting e processing may solve several problems, such as the decrease in seasonal labour availability. Picking soft fruits is still mainly performed manually, in order to avoid mechanical damage of the fruit. The cost of harvesting is a significant percentage of the total cost of fruit production. To realize an efficient mechanical harvesting, one of the major problems is the design of a proper gripping device, able to harvest and handle delicate objects, such as soft fruits, conforming to their various shapes and sizes.

Some fruit-picking mechanisms are based on a cutting device attached to a tubular arm [6]. The mechanism is provided with a rotating lip that detaches the fruit already enclosed in the tube; the fruit then rolls down the tube to a holding bin.

For picking delicate objects, as the agricultural products are, many applications employ the vacuum principle in order to realize a delicate gripping phase, avoiding high local pressure and damaging. Particular suckers or vacuum cups are frequently used when harvesting by means of vacuum techniques. The use of vacuum grippers, for picking fruits is described in [7, 8]. This method shows good results, mainly when attempting to pick tightly

---

Contact author: A. Manuello Bertetto <sup>1</sup>, C. Falchi <sup>1</sup>, R. Pinna <sup>1</sup>, R. Ricciu <sup>2</sup>.

<sup>1</sup>Piazza D'armi 1, 09123 Cagliari - Italy

<sup>2</sup>Via Santa Croce, 67 09123 Cagliari - Italy

clustered fruits, when the rotating-lip mechanism may have the tendency to put fruit away.

An interesting gripping device has been realised at Japanese University of Kyoto [9]. This gripper has three elastomeric fingers moved by McKibben muscles which allow the fruit to be wrapped.

At the Politecnico di Torino a soft hand for delicate fruit harvesting by flexible fingers pneumatically actuated was developed [10]. The hand has been designed so as to allow picking of quasi-spherical fruits having a diameter of about 50 to 150 mm. Particular versatility characteristics are required given the necessity of grasping objects of not constant size and shape which are not located along the hand axis. For this reason the hand is provided with the ability of adapting itself to the object location and size; in addition the hand has pneumatic force sensors to avoid object damaging, by controlling the grasping force. In [11, 12] a hand, designed at Politecnico di Torino, for the automatic harvesting of asparagus is described and characterised. This gripper allows a not centred grasping operation and can perform a fit back controlling the grasping force on the grasped object, by means of a customised contact force sensor.

A realisation of non-conventional pneumatics, which makes use of flexible actuators, is illustrated in [13]. In this work a small dimensions grasping hand for saffron flowers harvesting is described. This gripping device has one degree of freedom; it is moved by a deformable actuator which consists of a small bag made of elastomeric material, which is able to inflate and expand under the internal pressure action, when fingers are given an angular approaching movement.

An application of pneumatics in agriculture is illustrated in [14]. The objective of this system was to develop an automatic rolling plant for rice seedlings, which can roll up the seedling mat.

## 2 THE CROCUS SATIVUS CHARACTERISTICS

The *Crocus Sativus*, as represented in Fig. 1, is a specie of *Crocus* among the Iridaceae. *Crocus Sativus* flower bears the stigmas. From the stigmas that, are three and with a bright red colour is derived the spice. Saffron is certainly the world's most expensive spice by weight [15]. The Mechanisation of the harvesting, separation and drying of the stigmas is a suitable contribution to the quality and advantage of the Saffron growing [16, 17].

Saffron is graded via laboratory measurement of crocin (colour), picrocrocin (taste), and safranal (fragrance) content. Determination of non-stigma content and other extraneous matter are also key. Grading standards are set by the International Standard Organisation [18].

Saffron is actually produced in Europe, North of Africa, Iran, India, in Spanish and Greece; crops are also available from New Zealand, France, Switzerland, England, the United States, and other countries as indicated in Fig. 2.

Among the more high quality spice in the world, there is the Italian Saffron from L'Aquila in Abruzzo, Italian country, and from Sardinia, in the Mediterranean sea.

Despite the high number of mechanical and robotic machines, introduced in the agricultural field, the harvesting of flowers is not so much automated, not many devices exist for harvesting the really delicate Iridaceae, the family which the saffron belongs to [19, 20]. An automatic system for saffron flower harvesting has been realized at the Università degli Studi di Cagliari. In [21] a flower cutting system which preserves the young plant leaves is described. The system has been thought to be portable and operated by a worker.

The saffron harvesting begins when flowers start sprouting, during mid or late October, and lasts for a couple of weeks. Since the saffron is cultivated in open fields, the harvesting must be promptly carried out, in order to avoid crop loss if bad weather occurs, which would be disastrous for the blossomed flowers. It is also interesting to notice that a kilogram of saffron spice is made out of two hundred thousand flowers, leading to a cost per kilogram of about many thousand euros. Thus, it has been found worth to try and develop a device that would lead to the mechanization or semi-mechanization of *Crocus Sativus* harvesting, which would allow the productivity and work quality of pickers to be improved.



Figure 1 The *Crocus Sativus*.

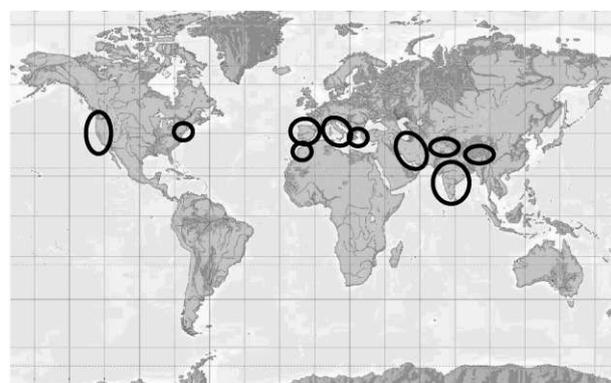


Figure 2 The Saffron cultivation zones in the world.

### 3 THE SAFFRON FLOWER HARVESTING DEVICE

The mechanical system to which this paper is concerned is a shoulder portable device with two main parts, the first of which is aimed at detaching the corolla from the stem and the second one is aimed at collecting the flower. As stated above, the detaching device has been specifically designed with the purpose of imitating one of the manual techniques adopted by the pickers.

The picker, after locating the flower and separating it from its leaves, holds the stem between its thumb and index fingers and, with a swift movement, moves it forward and downwards. The stem bends and breaks because of its frailty. In spite of leaves, which bend elastically and are much more flexible and tenacious, the stem strongly tends to brake when loaded. In fact, its structure collapses at the corolla basis, which is the area where the pickers hold the stem. This is mainly due to the morphology of the stem, which, from a macroscopic point of view, may be considered a thin long cylinder, axially loaded.

Actually, the spice production remains located strongly in an border role which not increase the economic importance of the product. The *Crocus Sativus* flowers must be harvested in the early hours of the day and then only during a 15- to 20-day period in November.

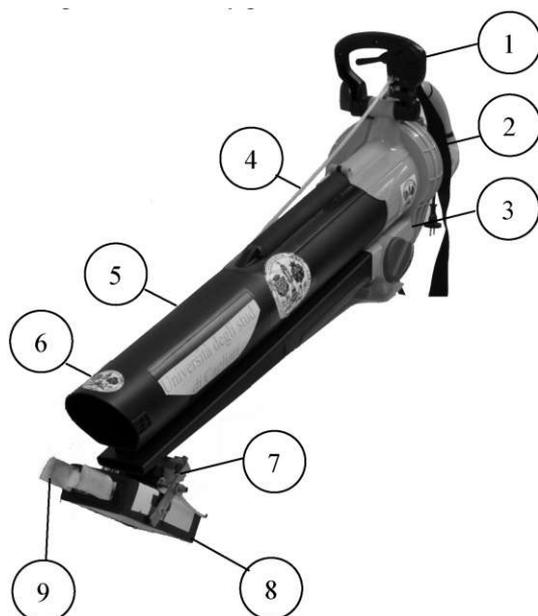


Figure 3 The portable device to harvesting *Crocus Sativus* on fields.

In Fig. 3 the portable device to harvesting *Crocus Sativus* on fields is represented. The device is an integrated light weight system with different parts allowing to realise the different phases that are necessary to perform an efficient harvesting operation on the field.

In particular in the Fig. 3 it is possible to see the handle (1), equipped by the pneumatic valve manually operated to perform the gripper pneumatic actuation, the body (2) supporting the electrical motor equipped by the fan to generate the vacuum to inhale the detached flower, the sponder (3), allowing to perform the harvesting without bending, the integrated pneumatic pipe to supply the

pneumatic actuator assembled in the gripper body, the vacuum tube (6) to collect the flowers, the link (7) to the device body of the gripper body (8), with the two fingers (9), to detach the saffron flowers. This link has a hinge elastically winded up around a horizontal axis to allow a rapid rotation of the gripper body when the lower is detached; this movement allows to approach the flower the vacuum system inlet.

The flowers are detached by means the gripper that is the end effector of the portable device to harvesting *Crocus Sativus* on fields. The gripper is a one degree of freedom pneumatically actuated gripper. The actuation of the gripper is pneumatic powered, in fact this kind of supply is very suitable to operate in agricultural fields.

In Fig. 4 it is possible to see an internal view of the gripper. It can be seen the aluminium alloy body (1) the pneumatic cylinder (2) with a diameter of 16mm and a stroke of 25mm, acting with its rod on a translating crossbar (3). This crossbar push two cylinder bodies (4) rototraslating in two bushing (5) where an helicoidal guide (6) drive a cam linked to the cylinders to obtain an helicoidally motion of the fingers (7) linked to the cylinders, the fingers detach the flower, realising a reciprocal rot translation without sliding each other.

The elastically powered hinge on the link indicated with (7) in Fig. 3 is loaded when the pneumatic cylinder moves the fingers.

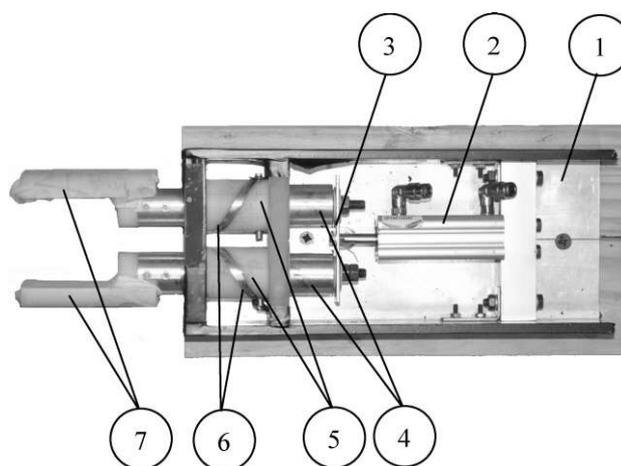


Figure 4 The gripper.

### 4 THE EXPERIMENTAL TESTS ON THE GRIPPER

The success of the detaching phase of the harvesting Saffron flower process is strongly dependent on the dynamic of the system. In fact, the sequence of the fingers movement and the bending rotation of the entire gripper body, around the elastically loaded horizontal hinge, must have a given time sequence. The harvesting procedure is efficient if the bending movement takes place when the fingers movement is completely done. This means that the cylinder dynamic must be sufficiently fast to have a sufficient strong shock at the finish of the cylinder stroke. In Fig. 5 the sensorized gripper is shown. The measured variables are the two pressures in the cylinder chambers,

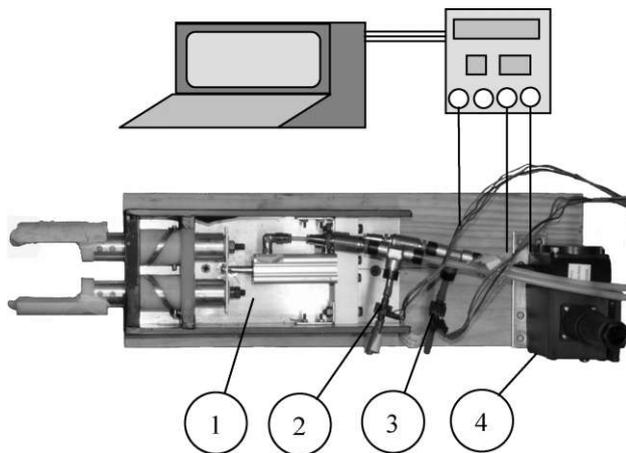


Figure 5 The experimental test rig.

the position and the speed of the cylinder rod, driving the fingers motion. In the same figure it can be seen the gripper described above (1), the pressure transducers (2) and (3) measuring the pressures in cylinder chambers and the wire position transducer (4). The transducers signal are processed by an acquisition board (Compact DAQ) National Instruments mod. NI 9219 with a resolution of 24 Bit and 100S/s/ch, and stored in a PC. The pressure transducers has a range of  $\pm 1.46 \times 10^6$  Pa with a sensitivity of  $6.89 \times 10^3$  Pa/mV; the wire displacement sensor has a 140mm range and a gain of 78.724 mV/mm.

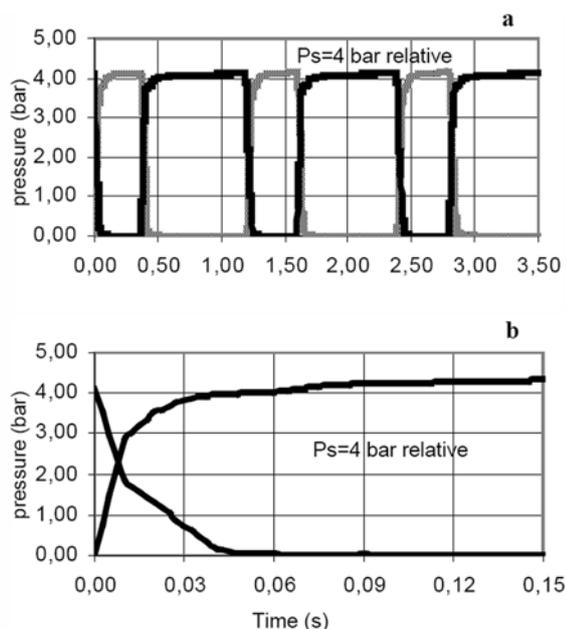


Figure 6 The pressure trend in the chambers at 4 bar relative pressure supply: many operations (a), the way out cylinder transient (b).

The graph in Fig. 6 refers the behaviour of the pressure values in the anterior and posterior chambers. The pressure value, when discharged to the ambient, has a trend with a

behaviour having a value quickly reaching the environment pressure level, also favoured by the decreasing volume for the piston movement. On the contrary, when the pressure increases in the cylinder chamber, the volume increases because of the piston motion and the pressure trend is more gradual.

Corresponding to the pressure values in the cylinder chambers, the position of the crossbar is referred in Fig. 7. The graph shows a rapid movement of the cylinder piston driving the fingers of the gripper. The curves are referred for a pressure supply of the cylinder of 1,5 and 4,0 bar relative. The graph trend has a loading ramp more steep, corresponding to the outstroke of the cylinder rod, compared to the trend of the in stroke. The area of the piston is, in fact, not symmetric: in the anterior chamber is present the rod. Observing the end of the outstroke displacement, it is possible to find an overshoot effect and the shock of the moving parts at the end of the stroke.

This fact allows the bending movement of the gripper round the horizontal hinge that represents the link to the system body. The cylinder piston velocity is referred in Fig. 8. The values are given for different supply pressures. The velocity values of the cylinder rod is rigidly linked to the velocity of the fingers in their helicoidally trajectories.

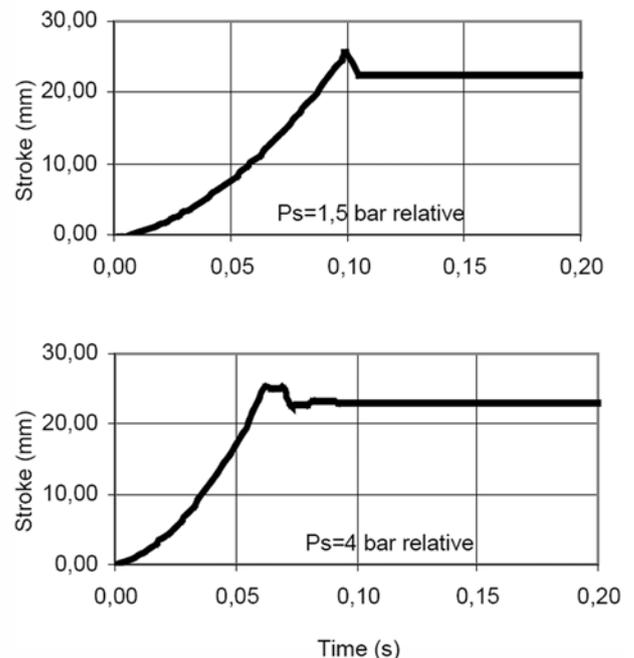


Figure 7 The cylinder stroke for different pressure supply  $P_s$ .

The velocity trend shows the influence of the supply pressure level: for a pressure supply of 1.5bar relative the velocity reaches the maximum speed value of 0.45m/s in 0.10 seconds for an acceleration medium value of 4.5m/s<sup>2</sup>. When the supply pressure is 4 bar relative, the maximum velocity value is 0.65m/s reached in 0.06 seconds; this corresponds to a medium acceleration of about 10m/s<sup>2</sup>.

The dynamic behaviour, strongly dependent on the supply pressure level of the two finger device, is responsible for the success of the harvesting operation. The authors think that it is important to know the gripper behaviour on field in operating conditions and not only in laboratory. A test procedure was then tuned in order to detect the kinematics and dynamic performance without hardware assembly of sensors on board. The gripper behaviour was then also studied by an image analysis technique.

This way of measurement has some advantages compared to that done by the wire displacement transducer performed on laboratory. By the image analysis technique it is possible to detect the displacement and velocity of the gripper fingers without an invasive assembly of the transducers. By this way is then possible to analyse the gripper behaviour on field where it is present humidity and environmental not so controlled conditions and, in addition, by this way any transducer is assembled on board. This is an important aspect when the gripper, assembled on the system, has also a bending motion as described when the finger gives the detached flower to the vacuum harvesting device.

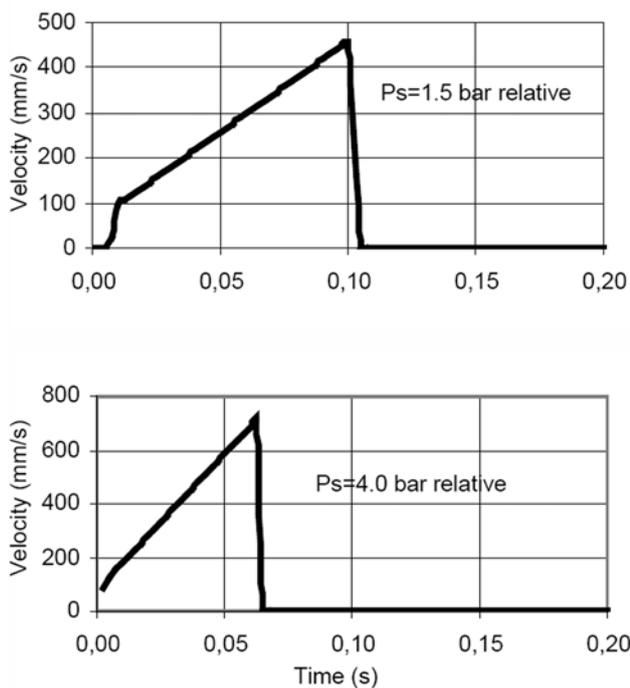


Figure 8 The cylinder velocity for different pressure supply  $P_s$

In Fig. 9 the test rig for image analysis techniques is shown. In particular it is possible to show one of the lamps (1) to illuminate the system under test, the unassembled integrated system (2) with the pneumatic valves to control the fingers motion, the air compressor (3) to supply the pneumatic cylinder driving the fingers, The gripper under test (4), the high speed camera (5) shooting the motion of

the triangle white target linked to the piston driving the gripper fingers and the acquisition system (6).

Image analysis technique was also applied in order to compute the pneumatic piston kinematics. A white triangle, with a lateral size of 40 mm, was connected to the piston and its movement was followed on a uniform black background. Ten cycles of the piston were recorded at 1000 fps by means of a high speed camera (Redlake Motionpro Y3) with CMOS sensor and a resolution of 1024 x 1280 pixels, equipped with a professional 50 mm lens. During the experiment, the apparatus was appropriately and uniformly illuminated.

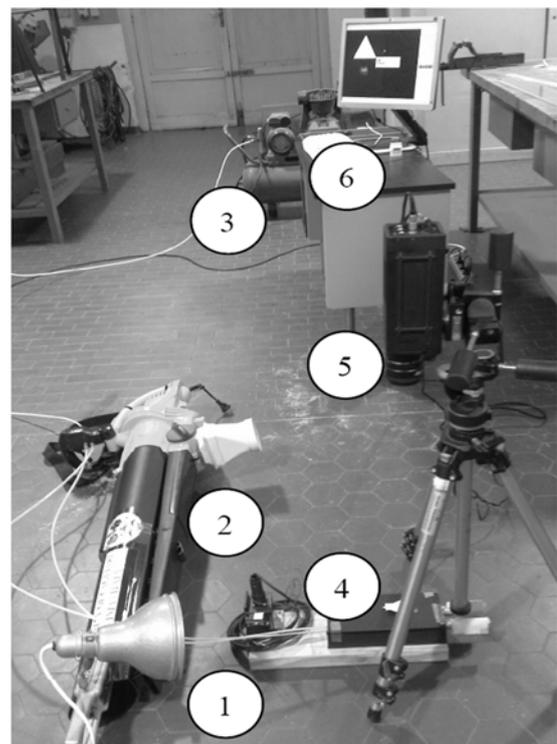


Figure 9 The test rig for image analysis technique.

Colour analysis was performed on the data: since the triangle is a white shape on a black background, applying an appropriate threshold allows identifying the triangle in the binarized images.

Consequently, the position of one of the triangle vertex is defined in each image. The displacements are then converted from pixels to mm using the size of the triangle as a reference measure. Hence, analysing the data with an in-house program code, the kinematics parameters of the piston are straightforwardly measured.

In Fig. 10a the gripper from the camera point of view is shown: it is possible to see the triangle white target linked to the piston driving the gripper fingers. One of the triangle vertex is defined in each image. This vertex is the higher vertex as indicated in Fig. 10b.

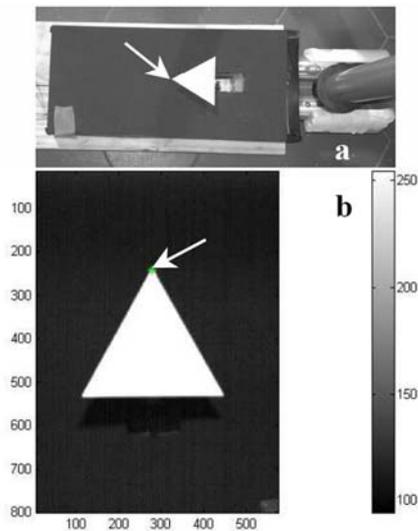


Figure 10 The test rig for image analysis technique.

In Fig. 11 the piston displacement is referred vs time detected by two gauges: the first one is given by the wire displacement transducer, the second by the camera.

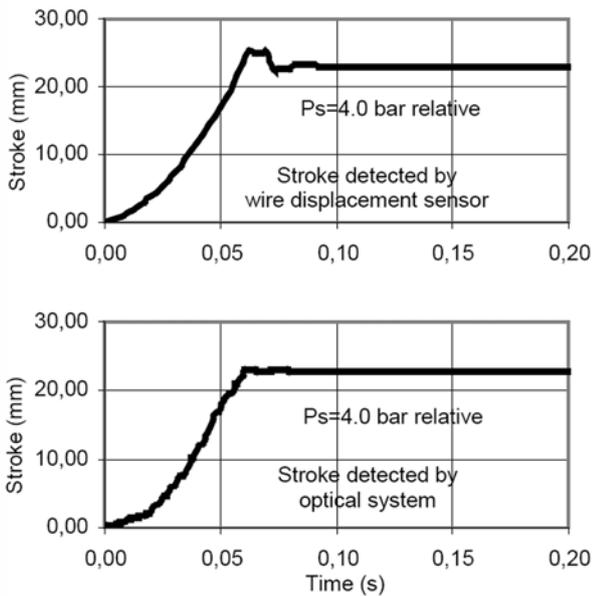


Figure 11 The cylinder stroke for pressure supply  $P_s$  of 4 bar relative, detected by wire displacement sensor compared to that detected by optical system.

As described above, the studied system is an integrated portable device with a vacuum collector. The vacuum system has a fan electrically supplied. This vacuum harvesting system generates a vacuum allowing an air flow suitable to the suction of the flowers picked by the two fingers gripper. During the tests the flow speed was evaluated by a differential pressure measurement to detect the air flow rate to harvest the detached flowers vs. the electric current absorption values.

The differential pressure was measured by a Pitot tube, placed in different radial positions, in the outlet duct of the vacuum harvesting system. In Fig. 12 the test rig to measure the flow rate to collect flower, when detached, is shown. It can be seen the device body (1) with the fan to generate the vacuum, the inlet duct (2) where the flowers are collected, the gripper (3), the outlet tube (4) to regularize the flow the multimeter (5) to measure the electric current absorption and the Pitot gauge to detect the dynamic pressure (6).

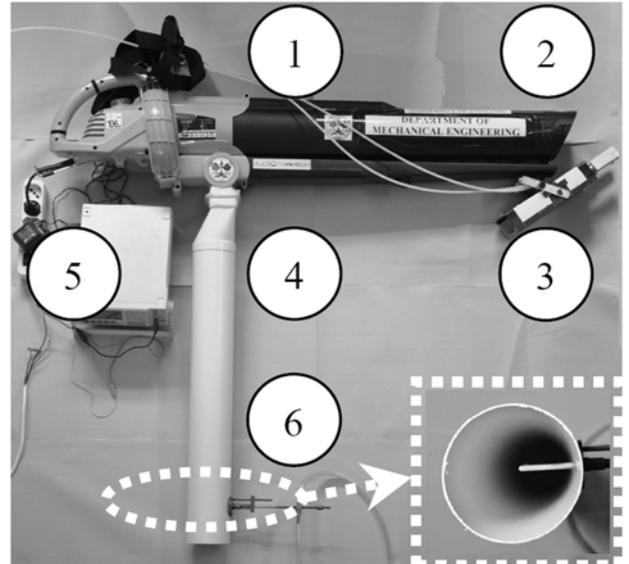


Figure 12 The harvesting integrated system with the outlet tube to regularize the flow and the Pitot tube gauge.

In the lower right side of the figure the axial view of the outlet tube shows the Pitot tube in the flow pipe.

In Fig. 13 the flow velocity, detected by the Pitot tube placed at different radial position respect to the outlet duct axis, is shown.

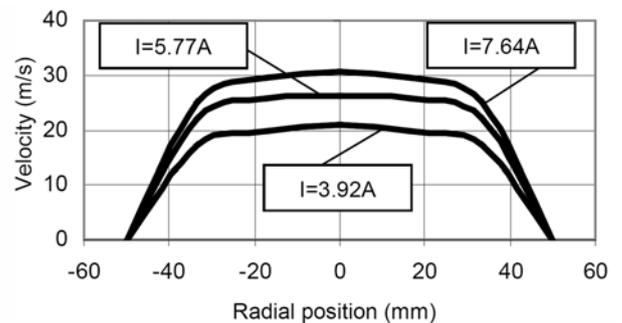


Figure 13 The flow velocity vs. the radial position, for different electric current absorption values  $I$ .

The velocity values are referred for three different electric current absorption values of the fan motor to generate the vacuum.

The flow rate referred vs. the electric current absorption in Fig. 14 was computed knowing the velocity values and the geometry section where the velocities are measured.

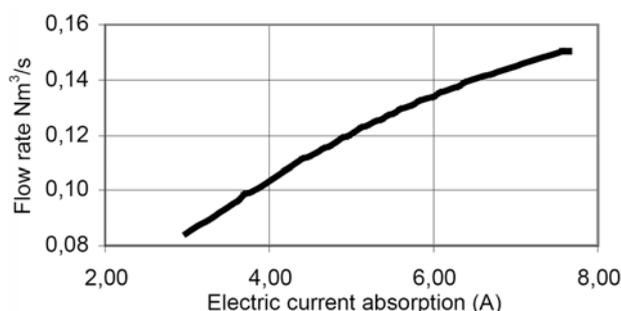


Figure 14 The flow rate of the suction vacuum arvesting device,vs. the electric current absorption value.

The graph shows a monotonic and regular growing of the flow rate level vs. the current absorption.

## 5 CONCLUSIONS

The mechanical integrated system for Crocus Sativus harvesting, described in this paper, was studied and characterised by laboratory tests. The tests was carried out on the prototype to highlight the dynamic behaviour of the detaching device, showing that it works as it was meant to and to know the vacuum system efficiency to harvest detached flowers. Experimental tests performed are useful to individuate the working parameters to obtain a dynamic sufficiently fast to perform an efficient detaching and harvesting strategy.

## ACKNOWLEDGMENT

The research activity was founded by Italian Ministry of Research and by Sardegna Ricerche. The authors would like to thank Dr Eng. Maria Grazia Badas for the contribution to the experimental tests.

## REFERENCES

- [1] Pilarski T., Happold M., Pangels H., Ollis M., Fitzpatrick K. and Stentz A., The demeter system for automated harvesting. *Autonomous Robots*, Vol. 13, No. 1, pp. 9-20, 2002.
- [2] Petrucci V.E., Clary C.D. and O'Brien M., *Grape harvesting systems principles and practices for harvesting and handling fruits and nuts*. Westport, CN: AVI Publishing, 1983, pp. 525-574.
- [3] Manfredi E., Mechanical harvesting tomatoes for processing. *Acta Horticulturae*, Vol. 277, pp. 63-68, 1990.
- [4] Giametta G., Mechanical harvesting of olives: present situation and prospects. *Acta Horticulturae*, Vol. 321, pp. 510-517, 1992.
- [5] Kondo N. and Monta M., Fruit harvesting robotics. *Journal of Robotics and Mechatronics*, Vol. 11, No. 4, pp. 321-325, 1999.
- [6] Harrell C.P., Adsit D., Pool T.A. and Hoffman R., The Florida Robotic Grove-lab. *ASAE Paper 88-1578*, St. Joseph, MI 49085, 1988.
- [7] Sarig Y., Edan Y., Katz N. and Flash T., Some aspects of robotics for fruit picking. *French-Israeli Bi-National Symposium on Advanced Robotics, Theory and Practice*, Tel-Aviv, May 30- 31, 1988.
- [8] D'Esnon Grand A., Rabatel G., Pellenc R., Journeau A, and Aldon M.J., MAGALI: a self-propelled robot to pick apples. *ASAE Paper 87-1037*, St. Joseph, MI 49085, 1987.
- [9] Namikawa K. and Ogawa Y., Study on the hand of fruit harvesting robot. *Research report on Agricultural Machinery*, No. 19, pp. 36-42, Kyoto University, Japan, 1989.
- [10] Carello M., Ferraresi C. and Visconte C., A new flexible pneumatic finger for a fruit-harvesting hand. *7th Int. Symp. On Fluid Control, Measurement and Visualization*, Sorrento, Italy, 2003.
- [11] Ferraresi C. and Manuello Bertetto A., Self-adaptive threefingered robot hand with tactile sensors. *Proceedings of the fourth International Symposium of Measurement and Control in Robotics*, pp. 275-280, Slovakia, June 12-16, 1995.
- [12] Ferraresi C., Manuello Bertetto A. and Mattiazzo G., A low-cost pneumo-electronic tactile sensor. *ICRAM 95*, pp. 1098-1103, August 14-16, Istanbul, Turkey, 1995.
- [13] Raparelli T., Beomonte Zobel P., Durante F. and Antonelli M., Dispositivi per la raccolta automatica del fiore dello zafferano, *Oleodinamica & Pneumatica*, Vol. 11, pp. 102-106, dec. 2006.
- [14] Yi-Chich Chiu and Din-Sue Fon, Development of an Automatic Rolling System for Rice Seedlings. *J. Agric. Eng. Res.*, Vol. 76, pp. 149-156, 2000.
- [15] Hill T., *The Contemporary Encyclopedia of Herbs and Spices: Seasonings for the Global Kitchen*, Wiley, ISBN 0-471-21423-X , 2004.
- [16] Ruggiu M. and Manuello Bertetto A., A mechanical device for harvesting Crocus Sativus (Saffron) flowers. *Applied Engineering in Agriculture, American Society of Agricultural and Biological Engineers (ASABE)*, Vol. 22, No. 4, pp 491-498, 2006.
- [17] Gracia L., Perez-Vidal C. and Gracia-Lopez C., Automated cutting system to obtain the stigmas of the saffron flower. *Biosystems Engineering*, Vol. 104, pp. 8-17, 2009.
- [18] ISO 3632-Saffron (Crocus Sativus Linnaeus).
- [19] Williams G.A., *Flower Harvesters*. U.S. Patent No. 4761942, 1986.
- [20] Kondo N. and Monta M., Chrysanthemum Cutting Sticking Robot System. *Journal of Robotics and Mechatronics*, Vol. 11, No. 3, pp. 220-224, 1999.

- [21] Erriu N., Manuello Bertetto A. and Ruggiu M., A two fingers device for saffron flowers harvesting. *Applied Mathematics and Mechanics series, Acta Technica Napocensis*, No. 49, Vol. II, pp. 45-50, 2006.

# AUTONOMOUS SENSOR NETWORKS FOR MONITORING INDUSTRIAL PROCESSES WITH APPLICATION TO MOBILE ROBOTS

Gokul Balakrishnan

Somashekhar S Hiremath

Indian Institute of Technology Madras

## ABSTRACT

The technology and processes associated with manufacturing have undergone a sea-change during the last few decades. Many of the hi-tech and hi-precision operations in manufacturing operations have robot to carryout operations that are routine, highly complex and critical, hazardous or of high-precision in nature. Robots basically have sensors using which the operation sequence is manipulated as programmed. Such sensors/robots when networked, provides flexibility and faster turn-around in manufacturing. This paper makes a study on the present developmental status and details the design features of self organizing sensor networks for automation and process monitoring. The goal is to define a model-based development process for large sensor-actuator networks with a focus on wireless communication. It also makes a survey on the present developmental status and details the design features of networking modular production systems using embedded wireless multifunctional sensor networks. It is intended to use this work as basis for future research work in the area of cooperative behavior of robots and also in the area of modular production systems and Robotics.

Keywords: Wireless Sensor Networks, Communication, Production System, Process Monitoring, Mobile Robot

## 1 INTRODUCTION

Applied technology is an important component of any scientific and engineering research and more so in the field of robotics. The applications of robot in day-to-day activities for industrial and time critical fail-safe operations have been increasing rapidly in the past years. Some of the broader areas are in agriculture, automotive manufacturing, construction, entertainment, health-care, laboratories, security and surveillance, military, mining, warehouse operations etc. Robot for industrial applications include critical and complex assembly operations which are of repetitive nature, performing routine operations such as pick-and- place, process control operations in nuclear and robotic arm in space to perform time-critical and fail-safe operations with minimum human intervention.

---

Contact author: Gokul Balakrishnan

Precision Engineering and Instrumentation Laboratory  
Department of Mechanical Engineering  
Indian Institute of Technology Madras  
Chennai-600036, INDIA  
E-mail: gokulb@smail.iitm.ac.in

Applications such as walking robot for human health care of elderly or handicapped people and for recreational activities are also on the increase.

Robot is basically a device employing sensors, electronics, computing and control features to carry out certain intended functions and falls in the realm of mechatronics.

The use of multiple sensors/robots enables networking, resulting in various industrial applications viz., material handling, material transfer, machine loading and unloading, spot welding, spray coating, assembly operations in sequence and inspection.

This paper makes a study on the design and development features of Self Organizing Sensor Networks for Automation and Process Monitoring. Presented in this thesis are the results achieved in design of the system architecture as well as its implementation in a prototypical use case. It also makes a survey on the present developmental status and details the design features of networking modular production systems using Embedded Wireless Multifunctional Sensor Networks.

## 2 LITERATURE SURVEY

Akyildiz *et al.* [1], Townsend *et al.* [2] and Miao [3] present a good introduction to sensor networks and applications. Distributed microsensor networks are used both in civil and military applications. Microsensors also provide a more flexible approach to industrial applications and enable better quality control. Riedel [4] describes self-organizing sensor networks that may operate autonomously over long periods of time in remote environments. Examples for wireless system architectures, network topologies and real-time environment testing are discussed by Hill [5]. Jeong *et al.* [6] analyze the suitability of sensor networks for industrial processes with the conclusion that a sensor network cannot meet all requirements of each application because of different characteristics of heterogeneous components.

Agre *et al.* [7] and Clare *et al.* [8] discuss the Self-organising in a microsensor is its capability to sense and process the information along with the adjacent ones and communicate. Such sensors may be required to operate autonomously over long periods of time in remote environments which calls for design for very low power. Park *et al.* [9] describes self-organizing sensor networks that may operate autonomously over long periods of time in remote environments which also requires a very low power design.

When such a system is configured for monitoring in a factory operation, it is essential that it shall have the attributes such as ease of operation, easy configurability/re-configurability, amenable to add-on stations, sensors and add to all these, ease of communication and operability by a factory worker. Thus, when implemented for industrial applications, the microsensor should have application specific communication and network protocols along with operating algorithms in order to ensure seamless connectivity and reliability under different conditions. An extensive treatise on the system architecture, network topology, the realtime environment testing including timing constraints, communication and security and system evaluation matrix are detailed in Collier *et al.* [10] and Nekoogar *et al.* [11].

The basic principle and design of various components of robot such as manipulators, end effectors are dealt with in detail in textbooks [12][13][14]. Intelligent autonomous robots find its application in many areas as explained in Sec. I. Justin [15] discusses the mobile robot/sensor networks that have emerged as tools for environmental monitoring, search and rescue, exploration and mapping, evaluation of civil infrastructure and military operations. Shibata *et al.* [16] describes the realization of a modular distributed control architecture specifically designed to control locally intelligent robot agents. On the vision algorithms, Jose & Giulio [17] discusses the implementation of a set of visually based behaviors for navigation.

A number of papers discuss on the communication network and control techniques. Firmansyah [18] has developed a prototype of modular networked robot for autonomous

monitoring with full control over web through wireless connection. The robot is equipped with a particular set of built-in analyzing tools and appropriate sensors to enable independent and real-time data acquisition and processing. The paper focusses on the microcontroller-based system to realize the modularity. Alberto Elfes [19] discusses a distributed control system that provides scheduling and coordination of multiple concurrent activities on a mobile robot.

Obstacles, motion control, logic and estimation are dealt in detail by the following authors. Yang Tian-Tian *et al.* [20] considered the problem of formation control and obstacle avoidance for a group of nonholonomic mobile robots. On the basis of suboptimal model predictive control, two control algorithms were proposed. Ralph P. Sobek *et al.* [21] presents a distributed hierarchical planning and execution monitoring system and its implementation on an actual mobile robot. The planning system was a distributed hierarchical domain independent system called Flexible Planning System (FPS). It is a rule based plan generation system with planning specific and domain specific rules. Luca Consolini *et al.* [22] dealt with leader-follower formations of nonholonomic mobile robots, introducing a formation control strategy alternative to those existing in the literature.

Pertaining to robot intelligence, Volker Turau [23] presented a model for a robot control system that is under development for use in a manufacturing cell, which consists of various assembly stations and a material storage system.

## 3 WIRELESS SENSOR NETWORKS

Today wireless sensor network forms the backbone of modern industrial production systems. The industries that call for implementation of sensors which are wireless, network with local processing at their respective nodes and synthesizing and display systems so that one can implement and monitor the whole operation from anywhere when implemented over internet using appropriate communication protocols.

## 4 PRODUCTION SYSTEMS

### 4.1 INTRODUCTION

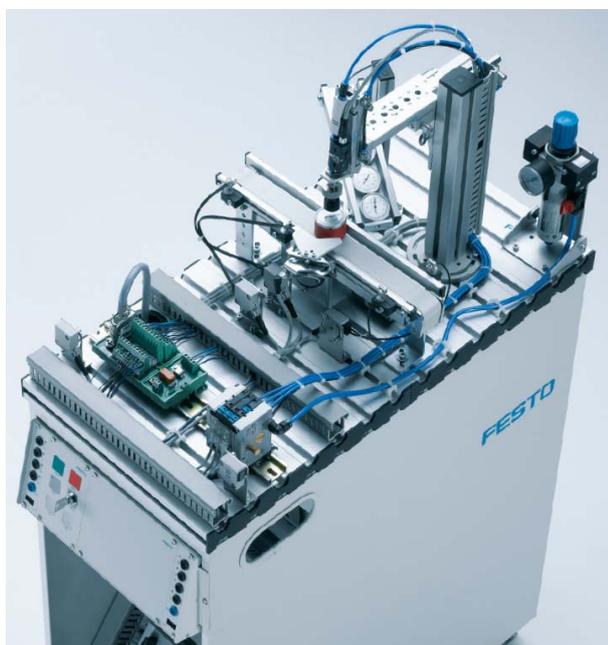
A new paradigm has evolved in the recent years in the field of production which is known as Modular Production systems. Modular Production System (MPS) is a learning system which can be used to simulate real-life industrial production systems of varying complexity. MPS is universal, modular, and open to further expansion. The basic level may consist of simple operations and sequences, which could be expanded step-by-step towards building a complex system.

Training on the actual production plant is often not feasible, as the risk of a system to malfunction would be huge and the production process would thus be significantly disrupted causing losses. A practice-oriented

training system is hence the optimum alternative to training in the work place. With this, trainees and skilled workers can be prepared to meet professional demands of a job without any pressure of time or facility constraint. In practice, a team consisting of industrial and electrical technicians would be able to assemble, commission, operate and maintain a production plant and at the same time using MPS, localise and eliminate any fault in a system through simulation without actually being part of the production facility.

#### 4.2 MPS DESCRIPTION

The support system, basic units and modules provide the fundamental building blocks, thus making the entire application package. The support system that consists of an aluminium profile plate as well as PLC board and trolley forms the central location for assembly of the basic units and the basic units perform individual functions, such as the picking up of workpieces. Useful functions can be carried out by using different modules such as a rotary indexing table and a drilling module. Modules and basic units are combined mechanically and via control technology to form stations which are capable of various functions. Several stations are available for standard functions such as distribution, testing, processing and storage. The functions of stations are summarised in Table 1.



(Courtesy: FESTO<sup>24</sup>)

Figure 1 The modular production system storage station.

MPS can provide the necessary support to manufacturer to tackle the challenges from the market competition as well as the shorter product life-cycle of the product. A typical module of a MPS is shown in Figure 1.

#### 4.3 MULTIPLE STATIONS

By means of a combination of different stations, it is possible to achieve installations with a wide range of functions. With these installations, one can deal with more complex training contents such as planning and project design, structured PLC programming techniques, operation of installations, commissioning and maintenance of Computer Integrated Manufacturing (CIM) systems, quality management, and industrial communication.

#### 5 SOFTWARE ARCHITECTURE

The wireless sensor network has many important components such as sensors, network, communication and software. The software architecture may be divided into four major layers namely:

1. Sensor or Machine Layer – The machine layer consists of the wireless sensor and the wired sensor. Data acquisition is carried out in this stage.
2. Network Layer – Data transport takes place through a sender/receiver unit and bus coupler, which are present in this layer. An optional media converter may also be added. Communication is the main aspect of this layer through self organization, wireless encryption and fault tolerance. Network management also plays an important role.
3. Data Processing/Cluster Layer – Data pre-processing take place in the cluster layer through the data pre-processing unit. Pre-processing is carried out for reduction of data volume and statistical analysis.
4. Control Station/Facility Layer – Data storage and presentation are the primary operations of the facility layer. The important components are structural database, Terminal for visualization and data backup unit. Data storage helps in association of data to objects such as transponders and RFID. Visualization helps in permanent monitoring at the operator's console. Typical software system architecture is shown in Figure 3.

#### 5.1 SENSORS

A sensor system suitable for these processes consists of statically mounted and of flexible placeable sensors. Using sensors with a wireless communication provides the flexibility to easily adapt to the network. Figure 2 shows the typical sensors and the network components. After the deployment phase it is easily possible to optimize the sensor coverage or to provide additional sensing resolution at specified locations for calibration tasks only by placing additional sensors there.

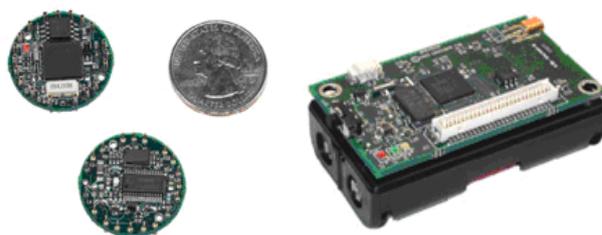


Figure 2 Embedded Sensors and Network components.

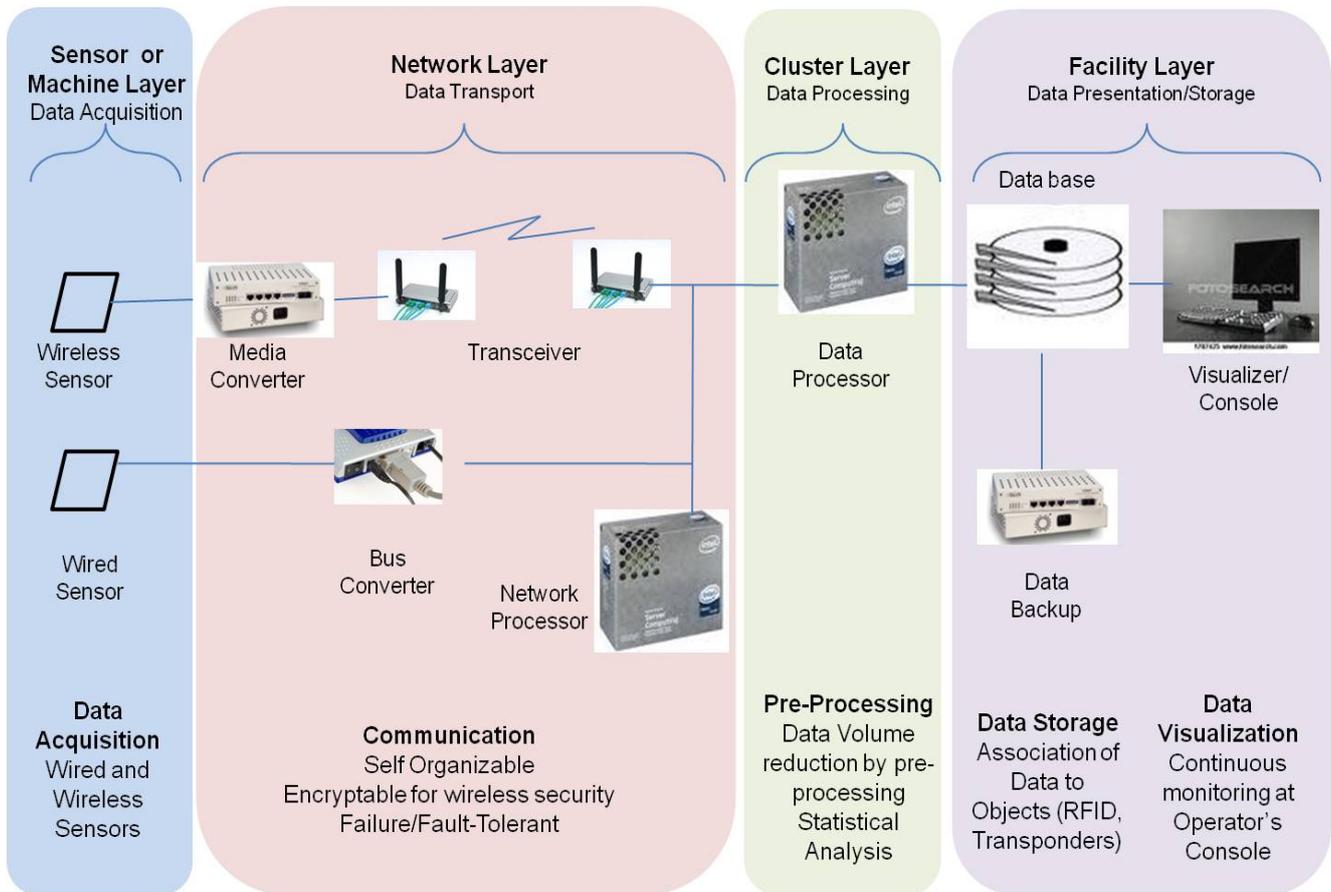


Figure 3 Software system architecture.

Table I - Different Functions of MPS Station

MPS Stations	Functions
1. Distribution	<ul style="list-style-type: none"> <li>To separate a workpiece from the rack.</li> <li>To make the workpiece available for subsequent processes.</li> </ul>
2. Testing	<ul style="list-style-type: none"> <li>Establish the material characteristics and check the workpiece.</li> <li>Either separate out a workpiece or make it available for a subsequent station.</li> </ul>
3. Processing	<ul style="list-style-type: none"> <li>Process the different workpiece types such as aluminium or plastic.</li> <li>Check the result of processing after the processing cycle.</li> </ul>
4. Handling	<ul style="list-style-type: none"> <li>Remove parts from the processing station.</li> <li>To sort parts according to their characteristics.</li> </ul>
5. Sorting	<ul style="list-style-type: none"> <li>Sort parts according to their characteristics.</li> </ul>
6. Buffer	<ul style="list-style-type: none"> <li>Transport and separate workpieces.</li> </ul>
7. Assembly	<ul style="list-style-type: none"> <li>Assemble a cylinder.</li> <li>Transfer the assembled cylinder to a subsequent station.</li> </ul>
8. Hydraulic punching	<ul style="list-style-type: none"> <li>Separate out a workpiece from the rack.</li> <li>Punch a hole in the workpiece.</li> <li>Rotate the workpiece by 180 degrees and transfer it.</li> </ul>
9. Functional testing	<ul style="list-style-type: none"> <li>Test the functioning of the assembled short-stroke cylinder.</li> <li>Either reject the cylinder or make it available to a subsequent station.</li> </ul>

## 5.2 SENSOR AND NETWORK REQUIREMENTS

In contrast to the cabled sensors, which have a constant power supply, the wireless sensors need to be powered by batteries. To increase the lifetime, energy harvesting techniques will be used. A further aspect to increase node lifetime is to reduce energy consumption. Usually the most power is needed for communication. To minimize the network traffic and thus power consumption, the sensor data will be pre-processed by the sensor nodes to minimize the size and number of the network package.

Facing these challenges, a new software architecture needs to be developed which is suitable for wired and wireless sensor nodes in the same way and which is easily implementable in industry processes.

To show the feasibility of this approach, a demonstrator is planned to be created and, in further steps of the work, this sensor system will be used in industrial production process for applications the industrial applications as stated above.

## 6 SYSTEM DESIGN, CONSTRUCTION AND TESTING OF THE PROCESSING STATION

The technical design aspects and methods planned to develop the proposed self organizing embedded sensor networks for process monitoring is described in this section.

After reviewing all the available stations, the processing station was selected to be explained in detail below.

### 6.1 DESIGN

The processing station performs a drilling task which can be broken into different sub-tasks. The functional description of the whole drilling process is summarised in Table 2.

### 6.2 ASSEMBLY AND COMMISSIONING

The components such as rotary table, drilling module, polishing module, sensors, etc. were gathered and assembled with reference to the system requirements.

When all components were in position, the commissioning process (cable connections) was then started. A PLC board was assigned to the station. The station was supplied by a 24V DC max. 5A power supply. The pneumatic components drew 6 BAR compressed air for working pressure.

### 6.3 PROGRAMMING WITH THE SOFTWARE TOOL

The stepwise refinement programming technique was applied to tackle the programming task. The programming of the complete operation was divided into five sub-modules, namely, (I) Rotary indexing table via a timer module, (II) Rotary indexing table via a sensor module, (III) Drill hole checking module, (IV) Drilling module, and (V) Indexing and drilling module. Finally, the program was completed by joining the above five sub-modules.

### 6.4 OPERATION AND TESTING

All five sub-modules were tested sequentially. Modifications were made from time to time in order to

fulfil the system requirements. Finally the complete operation was tested and the results were encouraging.

Table II - Functional Description

Start Condition	Initial Position	Sequence
<ul style="list-style-type: none"> <li>• Work-piece at the start of the conveyor belt</li> </ul>	<ul style="list-style-type: none"> <li>• Drilling unit in retracted position</li> <li>• Polishing unit retracted</li> <li>• Workpiece has reached the rotary table</li> </ul>	<ul style="list-style-type: none"> <li>• Rotate rotary indexing table</li> <li>• Check if drilling required</li> <li>• Extend drilling unit, Perform drilling operation</li> <li>• Switch off drilling unit, Retract drilling unit</li> <li>• Rotate rotary indexing table</li> <li>• Check for polishing</li> <li>• Perform Polishing operation if reqd.</li> <li>• Retract Polishing cylinder</li> <li>• Rotate rotary indexing table</li> </ul>

## 7 NETWORK ARCHITECTURE

This topic deals with the Network protocols that have been/to be implemented on the system. The concept is to control the MPS directly from the computer, without the use of the MPS Software. The idea of this project is to develop a self organizing wireless sensor network for process monitoring and automation. When this technology is transferred to the industry in future, it should be possible to easily integrate more sensors and microcontrollers to the sensor node.

The Final implementation would look as described in the Figure 4. The application would run on the host PC, which is connected to a gateway. The gateway would be connected to a switch. An ARM microcontroller and the MPS will also be connected to the switch. It is planned in a manner wherein the our network would use two protocols. The protocol between the host and the gateway would be a web services based protocol implemented through XML schema or SOAP or WSDL, as the idea is to make it compatible to Device Profile for Web Services (DPWS). A binary protocol has already been implemented for communication between the host and microcontroller, and also between the microcontroller and the MPS. The ARM microcontroller is programmed by C, but with a special uIP embedded code which is a special stack for 8/16 bit microcontroller programming. A C++ application is created on the host (PC) in order to map services and also to interact with the microcontroller.

The whole system can be divided into three important components namely:

1. Host – The host is the PC on which the application runs. The PC may also be connected to a database and a gateway. Initially, the gateway is planned to be used as the same PC or a different PC. Later on, a Fritz box maybe used.
2. Target – The target is an ARM microcontroller.
3. System – The system is the modular production system on which the developed protocols are tested.

### 7.1 DESCRIPTION OF THE MPS STATIONS

The first station is known as the storage station and the second station is known as the processing station. The storage station detects the colour of incoming workpieces and deposits them into one of the three levels of the rack module, depending on the colour (black, red, silver). A colour sensor detects the workpieces inserted onto the holder module and controls the further process sequence.

Detected workpieces are picked up by the pneumatic gripper and will be transported to the next free shell of the corresponding rack level via storage module. Each rack can accommodate six workpieces.

### 7.2 MINI AUTOMATION LINE

Two modular production systems are being used currently for testing purposes. The two systems have been connected together to form a mini automation line. Workpiece is first loaded manually into the storage station. Then the workpiece is accordingly loaded or unloaded by the robot arm gripper from the storage station and placed on the output conveyor belt. The workpiece then moves into the conveyor belt of the processing station. The processing of the workpiece takes place here and is then placed onto the output conveyor belt of the processing station. The workpiece is again moved to the storage station through the conveyor belts and hence completing the automation chain. This concept will be extended in future for multiple modular production systems.

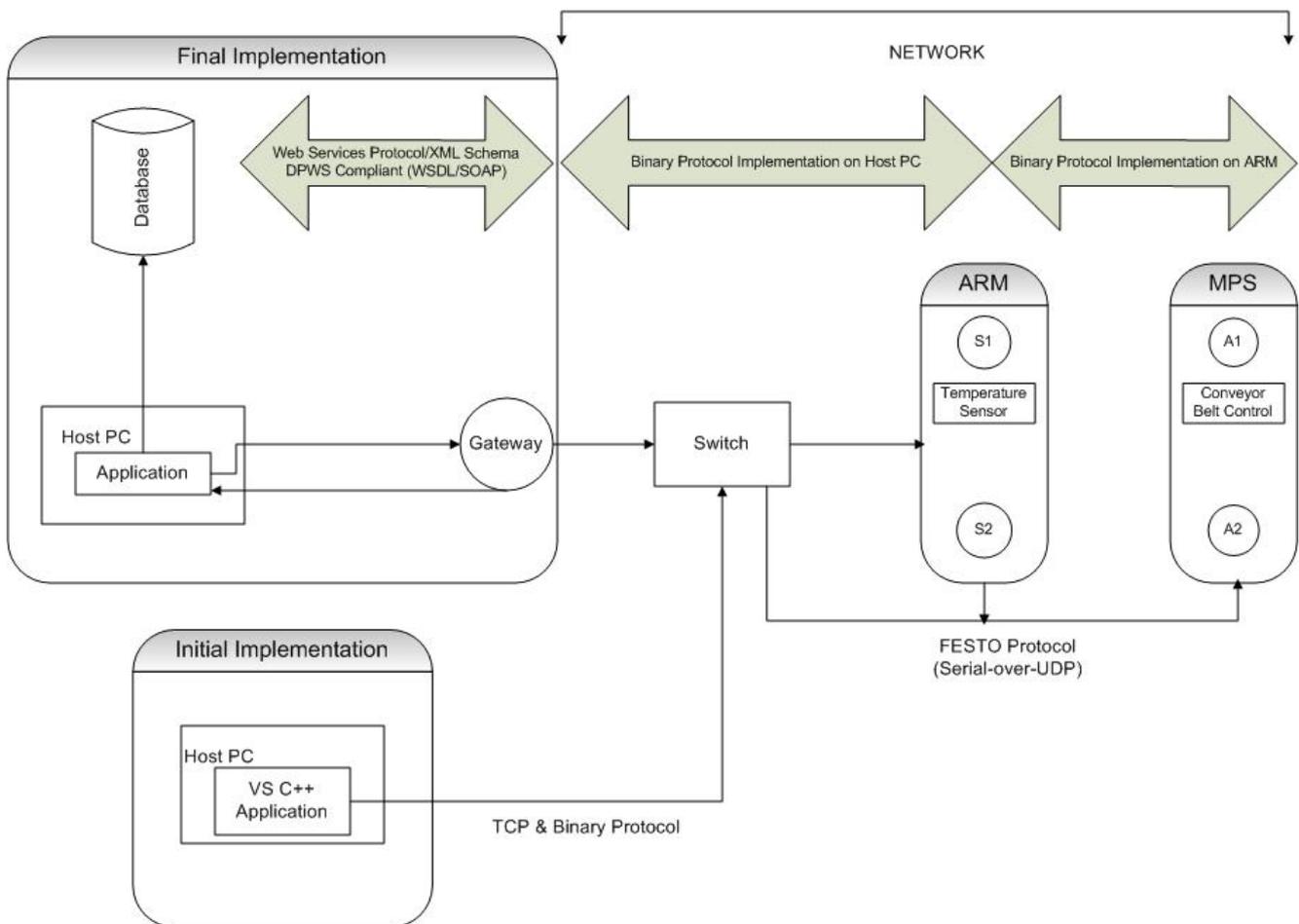


Figure 4 Network Architecture.

## 8 ANALYZING TIMING ISSUES IN PRODUCTION SYSTEMS

The timing aspects involved in the storing or loading process of the storage station is described in detail below in various steps. The sequence of steps is based on the program. The timings have been calculated based on actual experimentation and not assumed values. At the beginning, the program checks if the station has ever been reset. If the station is not yet reset (flag Reset\_OK = 0-signal) the program jumps to the "reset" sub module. If a work piece is available at the interchange (work piece holder between the conveyor belt and robotic arm), it is detected by the color sensor and put in the corresponding position on the rack.

### 8.1 ALGORITHM IN DETAIL ALONG WITH TIMING ASPECTS:

1. Initialization step
    - If "Reset\_OK" flag and "Init\_Pos" flag are set, continue with step 2.
    - If "Reset\_OK" flag is not set and Emergency-Stop is not active, jump to reset path.
    - Time for hardware reset (The robot arm performs a linear and rotary check) = 14s
  2. Interrogation of starting condition
    - If "F\_Start" flag is set, continue with step 4. (The start indicator light is not switched on)
    - If "F\_Start" flag is not set, continue with step 3: switch on start indicator light.
  3. Switch on the start indicator light (sends a request signal) and waits for starting condition to become true.
    - ("F\_Start" = 1-signal).
  4. Wait for a work piece at position P1. Time limit is not assumed, as in reality,  $t = 1$  to  $n$  is possible.
  5. When work piece is placed on conveyor belt at P1:
    - Sensor S1 detects the work piece, Time  $\leq 1$ s
    - Actuation process starts and conveyor belt moves.
    - Work piece moves and reaches position P2 on conveyor belt, time = 3s at a speed of  $v_1$  m/s.
    - Sensors S2 and S3 at position P2 detect the work piece, Time  $\leq 1$ s
    - Lever L1 pushes the work piece from position P2 to position P3, Time = 2 s
- $x_2$  and  $x_3$  are the distances on either side of the work piece.
- Robot arm operation starts
- Color sensor of robot arm detects work piece, time = 2s
  - "Station Occupied" signal received.
6. If Color type is black
    - Start movement, else jump to Step 13
  7. Operation
    - Open gripper,  $t < 1$ s
    - Extend gripper and go to fetch position, time = 2s
    - Close gripper, time = 1s
    - Load coordinates corresponding to the color,  $t < 1$ s

8. Move the gripper and place work piece in corresponding rack (black = rack C).
  - Time varies between  $t = 5$  to 10s (depends on the coordinate of the position to be placed such as A1 or B5 or C6 etc)
  - Retract gripper,  $t = 1$ s
  - Close gripper,  $t < 1$ s
9. Check if stock is full.
  - Cycle end.
10. Wait for reset button to be pressed.
  - If the reset button (control panel) or reset button (remote control) is pressed,  $t = 1$  to  $n$ .
  - Set Reset\_OK flag and continue with step 2.
11. If reset is not pressed
  - Drive actors (robot arm/gripper) to initial point
  - Move to initial position, time = 3-5s
12. If color type is red, continue with step 8, else jump to step 14.
 

Note the differences:

  - Color sensor recognizes different color.
  - Change in coordinates and hence positions red in B.
13. If color type is silver, continue with step 8.
14. If stock is full, make it empty manually and acknowledge.

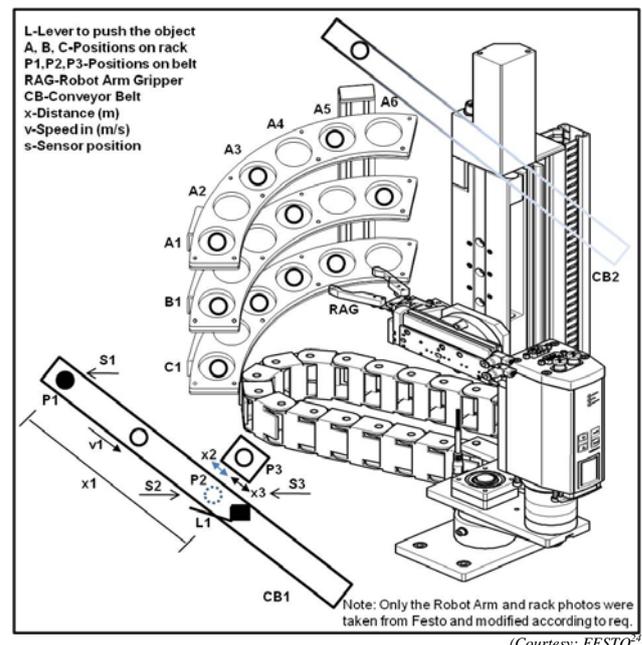


Figure 5 Modular production system block diagram.

At the beginning it is checked if a movement command is active. Corresponding to the coordinates, the controller outputs are set and it waits for "motion complete" signal each time.

Almost similar timing constraints are found to occur in:

- Retrieval/unloading process of the storage station
- One loading and unloading process of the storage station
- Drilling operation of the processing station.

## 9 PRESENT WORK

The presently ongoing work in this area aims to develop a robust and self organizing architecture for sensor networks for the process monitoring domain. The development process should allow the modular development of such systems, a seamless integration of further sensor nodes and follow a programming paradigm that we call 'a single system illusion'. The development of distributed real-time systems would become as easy as developing an embedded system consisting of one node. Hence, timing needs to be integrated into the modelling languages and the code generation would be automated. Furthermore, we need to abstract from the potentially heterogeneous network to provide total flexibility.

Once developed, in order to demonstrate the implementability, a production line consisting of various components is planned. The results will be used to optimize the manufacturing process, say for example, for tyres and reflow soldering plants.

## 10 IMPLEMENTATION

The autonomous self organizing sensor network setup would finally be linked to different modular production systems stations which are connected together to form an assembly line. This concept when developed would be used by various partners who participate in this work. For example, some plan to use the system for their production lines for soldering installation machines, others would use this system in their factories, and a service provider for nuclear plants are likely to implement this in a system treating nuclear garbage.

## 11 INTERFACING MOBILE ROBOTS

The design aspects and methods planned to develop the proposed Embedded system based Control for Networking mobile Robots (ECNR) is described in this section. The ECNR can aptly be an extension of the present work as an application. The Autonomous Wireless Sensor Network setup along with the Modular Production Systems will be linked to the Mobile Robot finally. The mobile robot will effectively pick the object from one of the Production Units and place it in the other unit. The interfacing of Mobile robots to the Wireless Sensor Network as well as the Production systems are elaborated here. Also discussed are the various Design Aspects which includes Hardware Setup, Embedded system based Control for Mobile robots and a suitable Software Architecture. An overview on the Vision Unit, Vision algorithms, Robot Kinematics and the Development Methodology are briefly described.

### 11.1 MOBILE ROBOT DESCRIPTION

Embedded Controller is the brain of the production system and is responsible for all actions. According to the function planning, the system hardware constitutes the following basic elements such as sensors, processor, controller, actuators, communication (user interface) and power

supply. Processor unit is responsible for processing the signal received from other modules and to send proper error signals back to them. Figure 6 shows the block representation of a mobile robot.

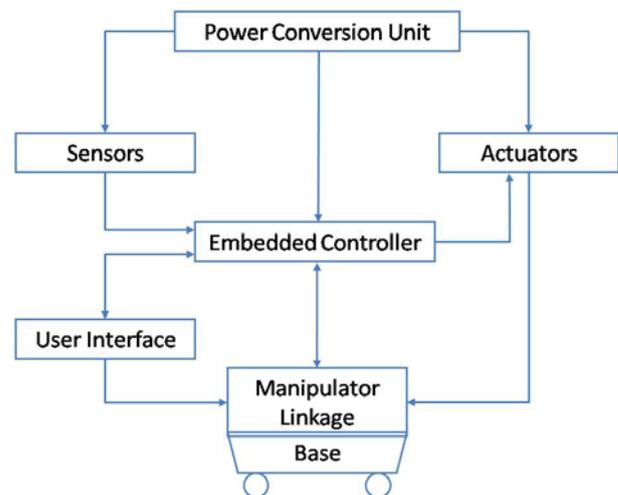


Figure 6 Block representation of a mobile robot.

Motor stepper driver (controller) magnifies the pulse width modulation from the processor to drive the motors. Sensors viz., proximity, colour, tactile, force and vision sensor etc are commonly used to sense the various signals and send them to the processor. The core of control system is either a DSP processor or microcontroller.

### 11.2 EMBEDDED CONTROL FOR A NETWORKED ROBOT

Embedded control configuration for a networked mobile robot is shown in Figure 7. It is configured around a server with a backup in order to provide the required fail-safe functioning during critical operations.

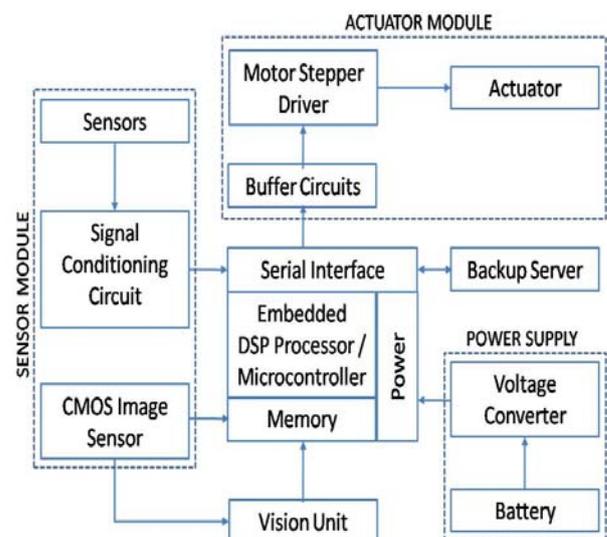


Figure 7 Embedded system based control schematic for networked mobile robot.

The server basically takes the load off the individual robots and all the main functions are deposited and controlled from the main server while the backup functions as redundant or hot-standby. The configuration and various elements are described in the paragraph above

### 11.3 CHOICE OF MOBILE ROBOTS

Some of the known mobile robots, viz., E-Puck and Amigobots provide necessary knowledge on motion planning and control strategies and are suitable for usage in development of an ECNR. The E-Puck mobile robot is preferred because of its miniature size and availability of multiple sensors and actuators matching the requirements of ECNR. While the Amigobot is expensive, it has the range capability for indoor as well as outdoor operations and has a built-in wireless module.

### 11.4 SOFTWARE ARCHITECTURE OF THE EMBEDDED SYSTEM

Typical software system architecture of an ECNR is shown in Figure 8.

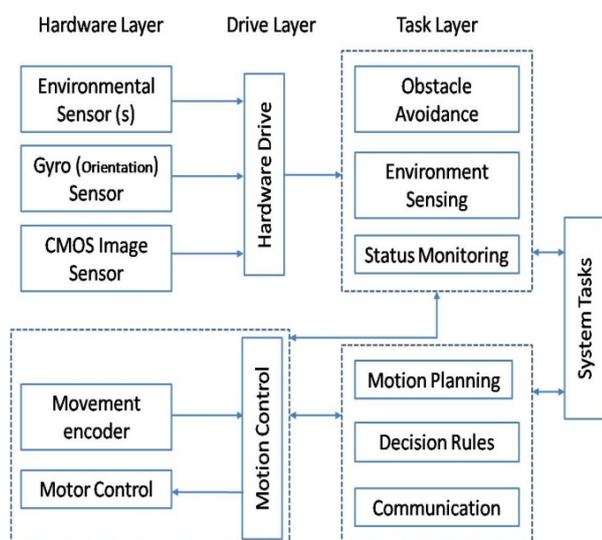


Figure 8 Software system architecture.

The drive layer interfaces the hardware and task layer. The hardware layer consists of various sensors for detecting the environment, path constraints, motion, speed etc. Gyro sensor gives the orientation and the CMOS image sensor provides the vision capability that is basic to mobile robots. The sensor information is processed by the various modules of the task layer, viz., in motion planning, rule based decisions and generating the correction/commands for communication to the motor control.

### 11.5 VISION UNIT

Vision being the complex of all the sensors, a brief description on the typical features is discussed in this section. This unit consists of a CMOS image sensor, dual-port RAM and necessary real-time features for communication with DSP. The sensor will have a normal sampling rate of 30-50 frames per second. This sensor has features for implementing the white equilibrium, exposure

control, saturation and tonality control. The image captured and stored in the RAM is used by the DSP during every sampling period. The processing involves reading of a static image at the target position and processing. Thus the system can make a judgment through the sampled data of several lines. Since the memory requirement for image storage is not very large, a high speed dual-port RAM is adapted.

### 11.6 VISUAL ALGORITHM

CMOS Image sensor provides RGB data. The hefts of RGB own a strong relation amongst each other and makes it difficult to process. HSI model is used for the object identification algorithm. It separates the intensity data from the colour information. Considering the system storage capacity associated with such operations, an evenly distributed line sampling detection needs to be implemented. The result is expected to give definite information about the block's position and its rotation angle in the horizontal direction. HSI model uses H heft as the main partition parameter. The basic image processing algorithm to obtain HSI parameters from RGB input are shown below.

$$I = \frac{1}{3}(R + G + B) \quad (1)$$

$$S = 1 - \frac{3}{R + G + B} [\min(R, G, B)] \quad (2)$$

$$H = \arccos\left\{\frac{[(R - G) + (R - B)] / 2}{[(R - G)^2 + (R - B)(G - B)]^{1/2}}\right\} \quad (3)$$

The processing algorithm should be simple enough to occupy less memory and fast enough but at the same time, capture the image features within the real time period.

## 12 ROBOT KINEMATICS

Robots movement involves both kinematics as well as dynamics. While details on these are considered out of scope of this paper, a small brief is provided here.

Robot movement consists of translation as well as rotation and these have to be calculated with reference to a fixed reference coordinate system in order to estimate the movement of every single robot. In a robot, each link and parent joint form their own local coordinate system. Rotation, translation and other transforms can be performed through matrix multiplication as depicted below.

Translation (D):

$$\begin{pmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 0 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4)$$

Rotation about x (Rx), y (Ry), and z (Rz) axes:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (5)$$

### 12.1 LOCOMOTION OF WHEELED ROBOTS

The locomotion of wheeled robots may be carried out in any of the following four standard methods:

- Differential drive, eg – Pioneer 2-DX
- Car drive (Ackermann steering)
- Synchronous drive, eg – B21
- Mecanum wheels, eg – XR 4000

### 12.2 POSITION ESTIMATION

The position estimation can be carried out by the ‘Dead Reckoning’ method as explained in Figure 9.

$$\Delta\theta = \frac{\Delta r - \Delta l}{d} \quad (6)$$

$$\Delta s = \frac{\Delta r + \Delta l}{2} \quad (7)$$

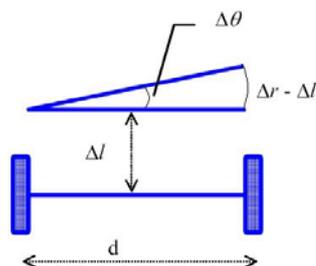


Figure 9 Position estimation using dead reckoning method.

## 13 CONCLUSION

This work when implemented can provide an easier access to parameter acquisition and monitoring in production environments; It would help to increase productivity and reliability and reduce energy consumption. The goal is to obtain multifunctional sensor networks for process data monitoring which should be achieved. Web service based network infrastructure would be developed with sensor networks having self organization capability and tool support providing modeling and integration capability for developing various industrial applications.

## ACKNOWLEDGMENT

Gokul, the author, would like to acknowledge his advisors at the Robotics and Embedded Systems Laboratory at TU Munich, Germany, who provided him with the exposure towards this conceptual work.

## REFERENCES

- [1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks”, *IEEE Comm. Magazine*, 2002.
- [2] C. Townsend and S. Arms, “Wireless Sensor Networks: Principles and Applications”, *Sensor Technology Handbook*, Vol 1, Chapter 22, pp. 575-588, 2004.
- [3] Y. Miao, “Application of sensor networks”, *Seminar on wireless self-organization networks*, Friedrich-Alexander-Universitat Erlangen-Nurnberg, 2005.
- [4] T. Riedel, “Self-organizing, wireless sensor network”, *Remote Site & Equipment Management Magazine, Millennial Net*. 2004.
- [5] J. L. Hill, “System Architecture for Wireless Sensor Networks”, *PhD thesis, University of California, Berkeley*, 2003.
- [6] W. Jeong and S. Y. Nof, “Performance evaluation of wireless sensor network for industrial applications”, *Journal on Intelligent Manufacturing*, Vol 19, pp. 335-345, 2008.
- [7] Agre J.R., L. P. Clare, G. J. Pottie and N. P Romanov, “Development Platform for Self-Organizing Wireless Sensor Networks”, *Supported by DARPA Tactical Technology Office*, 1999.
- [8] Clare L.P., G. J. Pottie and J. R. Agre, “Self-Organizing Distributed Sensor Networks”, *Supported by DARPA Tactical Technology Office*, 1999.
- [9] Park S., K. Shin, A. Abraham and S. Y. Han, “Optimized Self Organized Sensor Networks”, *Sensors*, ISSN 1424-8220, 2007.
- [10] Collier T.C. and C. Taylor, “Self-Organization in Sensor Network”, *Preprint submitted to Elsevier Science*, 2003.
- [11] Nekoogar F., F. Dowla and A. Spiridon, “Self Organization of Wireless Sensor Networks Using Ultra-Wideband Radios”, *Technical report by Lawrence Livermore National Laboratory*, 2004.
- [12] John J Craig, “Introduction to Robotics – Mechanics and Control”, Third Edition, *Pearson Education*, 2009.
- [13] Robert J Schilling, “Fundamentals of Robotics – Analysis and Control”, *Prentice Hall of India Ltd.*, 2007.
- [14] Chris Morgan, “Robots – Planning and Implementation”, *IFS Publications*
- [15] C.Justin, Rafael Fierro, “Mobile robotic sensors for perimeter detection and tracking”, *ISA Transactions*, Vol. 46, Issue 1, pp. 3-13, 2007.
- [16] Junichi Shibata, Genichi Yasuda and Hiroyuki Takai. “Implementation of a modular control system for multiple intelligent mobile robots.” *In Proceedings of the 3rd Conference on JSME Robotics and Mechatronics Symposia*, pp. 44-47, 1998.
- [17] S.V.Jose, S.Giulio, “Embedded visual behaviors for navigation”, *Robotics and Autonomous Systems*, Vol. 19, Issues 3-4, pp. 299-313, 1997.

- [18] I. Firmansyah, B. Hermanto and L.T. Handoko, "Control and Monitoring System for Modular Wireless Robot", *Proc. of Industrial Electronics Seminar*, Indonesia, 2007.
- [19] Alberto Elfes, "A distributed control architecture for an autonomous mobile robot", *Artificial Intelligence*, Vol. 1, No. 2, 1986.
- [20] Yang Tian-Tian, Liu Zhi-Yuan, Chen Hong, Pei Run, "Formation Control and Obstacle Avoidance for Multiple Mobile Robots", *Acta Automatica Sinica*, Vol. 34(5), 2008.
- [21] Ralph P. Sobek and Raja G. Chatila, "Integrated planning and execution control for an autonomous mobile robot", *Artificial Intelligence in Engineering*, Vol. 3, No. 2, 1988.
- [22] Luca Consolini, Fabio Morbidi, Domenico Prattichizzo, Mario Tosques, "Leader-follower formation control of nonholonomic mobile robots with input constraints", *Automatica*, Vol. 44, pp. 1343-1349, 2008.
- [23] Volker Turau, "A Model for a Control and Monitoring System for an Autonomous Mobile Robot", *Robotics*, Vol. 4, pp. 41-47, 1988.
- [24] Festo Official Website and their MPS Manuals
- [25] A.N. Das, D.O. Popa, P. Ballal, and F.L. Lewis, "Data-logging and supervisory control in wireless sensor networks," *Int. J. Sensor Networks*, Vol. 6, pp. 13-27, 2009.
- [26] F.L. Lewis, "Wireless Sensor Networks," in *Smart Environments: Technologies, Protocols, Applications, Chapter 2*, ed. D.J. Cook and S.K. Das, Wiley, New York, 2005.
- [27] A. Tiwari, F.L. Lewis, and S.S. Ge, "Wireless Sensor Networks for Machine Condition Based Monitoring," *Proc. Int. Conf. Control, Automation, Robotics, and Vision*, pp. 461-467, Kunming, China, 2004.
- [28] M. Bonfe and C. Fantuzzi, "Design and verification of mechatronic object oriented models for industrial control systems," in *Proc. IEEE Conf. Emerging Technol. Factory Autom. (ETFA)*, Vol. 2, pp. 253-260, 2003.
- [29] M. B. Younis and G. Frey, "Formalization of existing PLC programs: A survey," in *Proc. IEEE/IMACS Multiconf. Comput. Eng. Syst. Appl. (CESA)*, Lille, France, 2003.
- [30] V. Vyatkin and H.-M. Hanisch, "Verification of distributed control systems in intelligent manufacturing," *J. Intell. Manuf.*, Vol. 14, pp. 123-136, 2003.
- [31] G. Frey and L. Litz, "Formal methods in PLC programming," in *Proc. IEEE Conf. Syst. Man, Cybern.*, Vol. 4, pp. 2431-2436, 2000
- [32] Y. Koren, U. Heisel, F. Jovane, T. Moriwaki, G. Pritchow, H. Van Brussel, and A. G. Ulsoy, "Reconfigurable manufacturing systems," *CIRP Ann.*, Vol. 48, pp. 527-540, 1999.
- [33] Rogers, G.G., and Bottaci L. "Modular Production Systems: A New Paradigm for Manufacturing", *International Journal of Intelligent Manufacturing*, Vol. 8, Issue 2, Chapman & Hall, London, England, pp. 147-156, 1997.
- [34] R.W. Brennan, L. Ferrarini, J. L. M. Lastra, and D. V. Vyatkin, "Automation objects: Enabling embedded intelligence in real-time mechatronic systems," *Int. J. Manuf. Res.*, Vol. 1, pp. 379-381, 2006.
- [35] K. Thramboulidis, "Model-integrated mechatronics--Toward a new paradigm in the development of manufacturing systems," *IEEE Trans. Ind. Inf.*, Vol. 1, No. 1, pp. 54-61, Feb. 2005.
- [36] P. Ballal and F.L. Lewis, *Wireless Sensor Network Design*, National Technology and Science Press, 2009.
- [37] R. Jordan and C.A. Abdallah, "Wireless communications and networking: an overview," *Report*, Elect. and Comp. Eng. Dept., Univ. New Mexico, 2002.
- [38] M.R. Basheer, V.S. Rao, M.M. Derriso, Self-organizing wireless sensor networks for structural health monitoring, *Proc. SPIE Smart Structures and Materials: Modeling, Signal Processing, and Control*, pp. 515-525, 2003.
- [39] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, Vol. 38, 393-422, 2002.
- [40] C.-Y. Chong and S. P. Kumar, "Sensor Networks: Evolution, Opportunities, and Challenges," *Proceedings of the IEEE*, vol. 91, 1247-1256, 2003.
- [41] C. Prehofer and C. Bettstetter, "Self-Organization in Communication Networks: Principles and Design Paradigms," *IEEE Communications Magazine*, Vol. 43 (7), pp. 78-85, 2005.
- [42] Q. Li, J. Aslam, and D. Rus, "Distributed Energy-Conserving Routing Protocols for Sensor Networks," in *Proceedings of the Thirty-Sixth Hawaii International Conference on System Sciences*, 2003.
- [43] Zhao F, Guibas L. *Wireless Sensor Networks: An Information Processing Approach*, Morgan Kaufmann: San Francisco, CA, pp. 240-245, 2004.
- [44] A. Tiwari, P. Ballal, and F.L. Lewis, "Energy-Efficient Wireless Sensor Network Design & Implementation for Condition Based Maintenance," *ACM Trans. on Sensor Networks*, Vol. 3, No. 1, pp. 1-23, 2007.
- [45] A. R. Sardesai, O. Mazharullah, and V. Vyatkin, "Reconfiguration of mechatronic systems enabled by IEC 61499 function blocks," presented at the *Australasian Conf. Robot. Autom.*, Auckland, New Zealand, 2006.
- [46] Somashekhar S H, Gokul B, "Embedded System based Control for Networking Mobile Robots", *International Conference on Mechatronics and Information Technology (ICMIT)*, Korea, pp. 86-87, 2009



# COLLISION AND PROXIMITY AVOIDANCE FOR ROBUST BEHAVIOUR OF REAL-TIME ROBOT APPLICATIONS

Alexandru Dumitrache

Theodor Borangiu

University Politehnica of Bucharest, Romania  
Centre for Research & Training in Robotics and CIM (CIMR)

## ABSTRACT

This paper proposes a set of techniques for predictive collision avoidance which ensure robust operation of robot applications. Implementation issues for applying the techniques to current state-of-art robots are presented, including integration with manual and automatic operation modes. Simulation results are also presented.

Keywords: Collision detection, robust operation, accidental damage prevention, robot motion prediction

## 1 INTRODUCTION

Collision detection, also known as interference detection or contact determination, is a fundamental tool in computer games and animation, physical simulation robotics and computer-aided design and manufacturing (CAD/CAM). In these applications, solid bodies are not allowed to penetrate each other; here, collision detection automatically reports a geometric contact when it is about to occur or has already occurred.

This problem has been widely studied, and there are many algorithms available for collision detection: from general algorithms dealing with arbitrary polygonal meshes with no particular structure (polygon soups), and particular algorithms, which exploit properties such as convexity or temporal coherence for faster queries.

Usually, the concept of collision detection encompasses the following elements:

- *proximity detection*, which refers to the minimum distance between two solid bodies;
- *collision detection*, which detects whether two solid bodies touch each other;
- *collision response*, which computes the changes in motion of the solid bodies after a collision.

In 3D Computer Aided Design (CAD) applications, the following queries are usually performed:

- *clash* (intersection) detection: checking whether two solid bodies intersect each other;
- *tolerance verification*: detecting whether two objects are closer than a given tolerance;
- *distance computation*: computing the minimum distance between two objects.

Traditionally collision detection (CD) was discrete: it only tested overlapping between two static instances of moving objects; however, CD routines might ignore collisions between two fast moving objects (e.g., they may not notice a bullet passing through a narrow wall). In contrast, continuous collision detection techniques (CCD) are guaranteed to find the collision which has occurred between two given static instances of the 3D scene, although they require more processing power than CD.

Recent research efforts concentrate on optimizing continuous collision detection, and also on applying it to deformable objects, which is useful for more realistic simulations. [1,2,3].

Two older surveys are available in [4] and [5]; however, they only present non-continuous collision detection methods.

---

Contact author: Alexandru Dumitrache<sup>1</sup>, Theodor Borangiu<sup>2</sup>

<sup>1</sup>E-mail: alexandru.dumitrache@cimr.pu.ro

<sup>2</sup>E-mail: theodor.borangiu@cimr.pub.ro

## 2 CASE STUDY: REVERSE ENGINEERING PLATFORM

Collision avoidance techniques were studied in the context of a reverse engineering platform (Figure 1 and 2), which integrates:

- a short range laser scanning sensor;
- a 6-DOF vertical robot arm, which sweeps the laser sensor around the scanned part;
- a rotary table (1-DOF), which holds the scanned part;
- a 4-axis CNC, able to create 3D parts from raw stock.

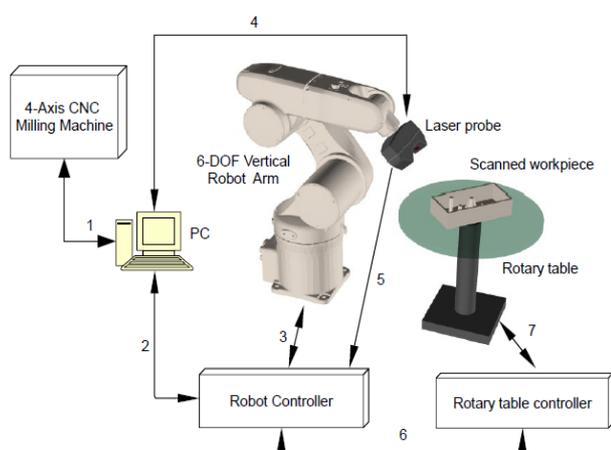


Figure 1 Overview of the reverse engineering platform.

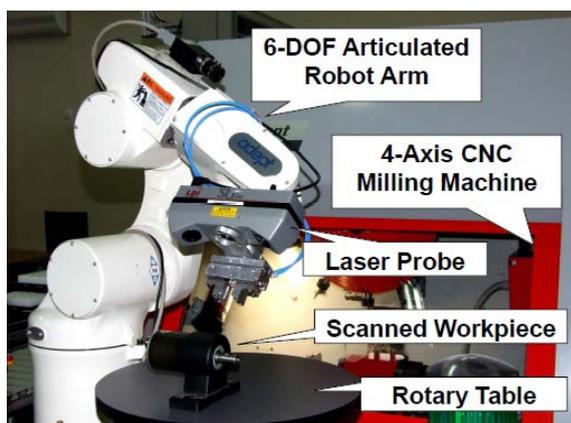


Figure 2 The 3D scanning system overview.

The scanned part can be placed on the rotary table; therefore the scanning system is a redundant mechanism with 7 degrees of freedom (DOF). The extra DOF is used for optimal planning the motion of the rotary table, satisfying the following criteria:

- ensure a smooth motion of the rotary table, in order to allow the part to be placed without any special fixture requirements;
- avoid singular configurations in the robot arm, which would cause high joint speeds, low accuracy in scanning and unexpected end-effector movements;
- avoid collisions between the following elements:
  - laser probe with robot arm itself (Figure 3);
  - robot arm and laser probe, with rotary table;
  - robot arm and laser probe, with the scanned part;
  - robot arm and CNC milling machine.
- minimize scanning time.

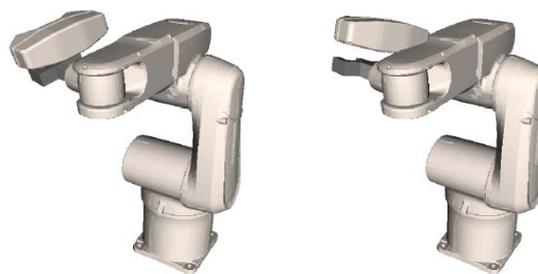


Figure 3 Possible collision between laser sensor and robot body.

In this scenario, collision detection is required at the motion planning stage, in order to make sure the robot motion, which is computed automatically by a planner, does not collide with any of the objects in the robot environment. The motion planning can be performed either offline, for relatively short motions, or in real time if the motion is longer and planning would take several seconds to compute.

Real-time collision detection can be also employed during the use of the laser scanning system. In this way, the collision detection engine acts as an extra layer of protection, preventing accidental damage from erroneous user input or even from a programming mistake in the scanning strategy.

Finally, the CAM software component, which converts a 3D surface model into a set of CNC milling toolpaths, has to compute collision-free safe toolpaths, without the risk of damaging the milling cutter or the clamping fixtures. Since NC paths are generated as a series of 2D steps, collision avoidance is integrated into the toolpath generator without explicitly making queries of 3D rigid body intersections. An algorithm for generating collision-free milling paths with tool engagement control for arbitrarily complex raw stock and part geometry was proposed in [6].

### 3 COLLISION DETECTION FOR ROBUST OPERATION

The 3D scanning module can be operated in two modes, which are not specific to this particular application, but employed in virtually all robotic tasks:

- From the robot's manual control pendant (MCP);
- In automatic mode, where scanning trajectories are generated by the robot control software.

In manual mode, the robot is usually moving at low speeds and the user is assumed to be careful not to cause collisions. However, a robust user interface shouldn't rely on correct user input; it should not allow the user to produce damage to the system no matter what the user input might be.

In automatic mode, the robot moves along parameterized scanning patterns (Figure 4), which also depend either on the user input, who may specify approximate part size, number of passes or scanning speed, or may rely on autodetection. However, users may make mistakes, and autodetection may produce incorrect results. It is difficult to predict when a certain combination of scanning parameters will lead to a collision or not; therefore, even after checking that all the parameters retrieved from user input are in the correct range, there is still possible that the scanning parameters may make the robot collide with the nearby equipment.

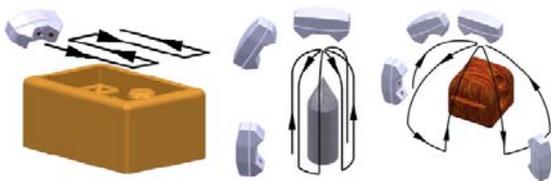


Figure 4 Predefined scanning patterns: grid, cylindrical and spherical.

The robot controller supports very limited collision detection primitives, and only checks the end-effector point against maximum 4 obstacles, which can have one of the following shapes: axis-aligned box (AABB), cylinder, sphere or frustum [7]. These obstacles can only be specified in a robot configuration program, and cannot be changed while a user application is running on a robot.

These limitations make it very difficult to define obstacles which enclose tightly the robot, sensor and table in order to provide robust operation with respect to collisions.

#### 3.1 COLLISION DETECTION DURING MANUAL OPERATION

The proposed protection scheme is to have a dedicated task for real-time collision checking during robot operation. The protection task runs on the PC, continuously monitoring the robot position and velocity.

In manual mode, no user program is allowed to move the robot or change its speed, due to internal protection mechanisms implemented in the robot controller.

When the robot is heading to a colliding situation, the only actions that could be taken from a user program are:

- Give visual feedback on the MCP;
- Give audible feedback to the user;
- Assert the emergency stop signal (only in extreme cases).

The first option is useless if the user is not looking at the MCP, which is the most likely situation. The second option uses the control pendant's internal speaker to emit warning beeps, which change in intensity as the robot approaches an obstacle. If a collision is imminent, the only action allowed by the robot controller is to assert the emergency stop signal. An overview of the protection scheme is given in Figure 5.

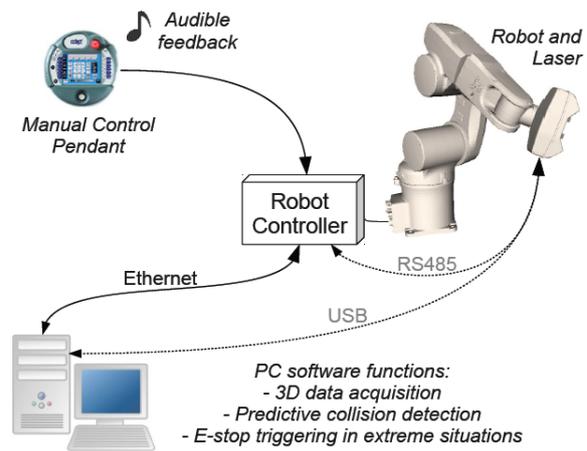


Figure 5 Protection scheme for collision-free manual mode operation.

A more useful approach would have been to automatically reduce the robot speed when a collision is imminent; however, in order to implement this behavior, safety mechanisms from the robot controller would have to be bypassed.

Warnings should be given only when two possibly colliding bodies become closer. When the user moves the robot away from the colliding situation, no warning should be given. To implement this, the protection module should know also in which direction the robot is moving. User programs do not have access to the buttons pressed on the control pendant; this information is accessible only inside the MCP. However, in manual mode it is relatively easy to predict the robot motion with good accuracy, since the robot can be moved in a single direction at a time. The possible directions are:

- Cartesian translation (any direction in 3D space);
- End-effector rotation (around any fixed axis);

- Joint motion (rotate only one robot joint at a time).

Therefore, a predictive collision detection mechanism can be implemented, as illustrated in the example from Figure 6. The predictor has to detect the type of motion:

- A joint interpolated motion
- A Cartesian motion (translation and/or rotation)

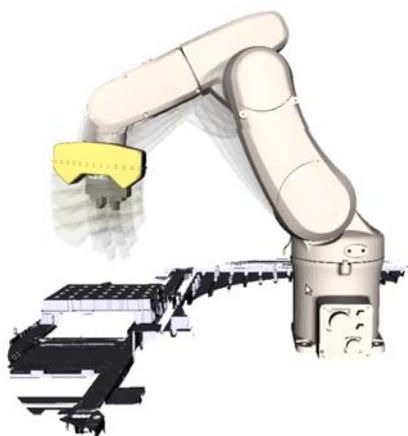


Figure 6 Prediction of a joint-interpolated robot motion in a virtual workspace.

A model for describing and predicting joint-interpolated motions is given by:

$$J_i^{(t+1)} = J_i^{(t)} + s \delta j_i, \quad i = \overline{1, n} \quad (1)$$

where the robot has  $n$  independent joints,  $J_i^{(t)}$  is the absolute position of  $i^{\text{th}}$  joint at time  $t$ ,  $\delta j_i$  are weights which represent the relative speed of the  $i^{\text{th}}$  joint and  $s$  is the speed factor. This model represents general joint interpolated motions; however, in manual mode, only one joint is moving at a time. However, due to vibrations, all the joints will appear to be moving, but only one of them with a significant amount (let's say joint number  $k$ ).

Mathematically,  $|\delta j_k| \gg |\delta j_i|, i \neq k$  for a fixed  $k$ . However, this does not have any significant negative effects in the prediction process.

Therefore, the model parameters are  $[\delta j_i], i = \overline{1, n}$ . These parameters remain constant throughout the motion and can be identified by nonlinear minimization. In contrast,  $s$  may change freely throughout the motion due to acceleration and deceleration.

Cartesian motions are described by linear interpolation in  $X$ ,  $Y$  and  $Z$ , and spherical linear interpolations (slerp) in orientation. Therefore, the rotation axis remains constant throughout the Cartesian motion. A model for predicting linear motions in Cartesian space, where the end-effector is allowed to change its orientation, is:

$$X^{(t+1)} = X^{(t)} + s \delta x \quad (2)$$

$$Y^{(t+1)} = Y^{(t)} + s \delta y \quad (3)$$

$$Z^{(t+1)} = Z^{(t)} + s \delta z \quad (4)$$

$$\theta^{(t+1)} = \theta^{(t)} + s \delta \theta \quad (5)$$

$$R^{(t)} = R_{[rx, ry, rz]}(\theta^{(t)}) \cdot R^{(0)} \quad (6)$$

where  $(X, Y, Z, R)$  is the Cartesian end-effector position and orientation ( $R$  is a  $3 \times 3$  rotation matrix),  $R^{(0)}$  is the initial end-effector orientation (at  $t = 0$ ), and  $[rx, ry, rz]$  is the rotation axis throughout the motion, which is constant. The notation  $R_{[axis]}(angle)$  is a rotation matrix specified by its axis and angle.

The model parameters, which remain constant throughout the motion, are  $[\delta x, \delta y, \delta z, \delta \theta, rx, ry, rz]$ . These parameters remain constant throughout the motion and can be identified by nonlinear minimization.

The decision for the motion type (Joint or Cartesian) is taken by trying to fit both models and select the one which gives lower residuals.

Transformations between Cartesian and Joint spaces are given by direct and inverse kinematics functions.

### 3.2 COLLISION DETECTION DURING AUTOMATIC OPERATION

Even if the trajectory planner is programmed to generate collision-free paths, nobody can be sure that there are no programming mistakes in the robot application software. In semi-automatic modes, where the robot moves between points taught by the user, trajectory planning is performed solely on the robot controller, which does not check for collisions; however, collision checking can be done before sending a motion instruction to the robot. Of course, this assumes the robot program runs from the PC terminal.

The collision detection mechanism described in this section is designed to be as general as possible, in order to be useful regardless of the particular robot application. The implementation is a watchdog task, which analyzes the subsequent motion transparently, while the program is running. If a collision becomes imminent, the following actions can be taken:

- User feedback (visual or auditive)
- Gradually reduce monitor speed (this can be done even while another program is running)
- Trigger the emergency stop (only in extreme cases)

In automatic operation, only one program is normally allowed to move the robot. However, there may be additional program tasks which can watch the robot motion, i.e. read the current position within a loop, and also retrieve the destination of the current motion. Therefore, the watchdog task knows in advance the robot trajectory, and no prediction is necessary. A schematic view of the protection scheme for automatic operation is given in Figure 7.

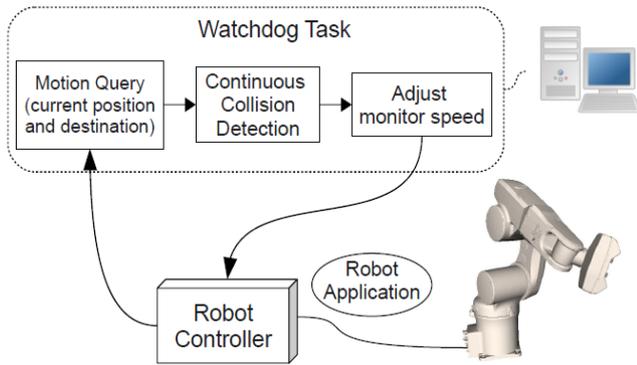


Figure 7 Protection scheme for collision-free automatic operation,

### 3.2.1 Experimental results

Robot position query can be performed from PC terminal every 32 milliseconds via Ethernet. A major cycle of the robot controller has 16 milliseconds. However, when network delays occur, the PC would not have the possibility to slow down the robot quickly enough. Therefore, a protection mechanism has been employed: if the time since last message received from the watchdog is higher than normal, the monitor speed on robot decreases gradually, even until a full stop if network delays are very high. However, this method is not yet reliable for high speed operations.

## 4 COLLISION DETECTION FOR PATH PLANNING

A heuristic planning algorithm for the robot arm and rotary table was presented in [8]. The 7-DOF mechanism (robot + rotary table) is redundant; this redundancy can be exploited to satisfy additional constraints. The angle of the rotary table,  $\theta_R$ , may be chosen freely; once this angle is fixed, the remaining 6-DOF can be uniquely chosen from the possible inverse kinematics solutions.

The algorithm provides support for additional constraints specified as real-valued functions  $f_i$ , with  $0 \leq f \leq 1$ , which evaluate any static robot pose, with the following meaning:

- $f_i = 0$  : the constraint is not satisfied
- $f_i = 1$  : the constraint is fully satisfied
- $0 < f_i < 1$  : the constraint is only partly satisfied

If there are many constraints, their functions can be multiplied, resulting a metric for evaluating any static robot configuration, with the same interpretation of values:

$$f = \prod_{i=1}^m f_i \quad (7)$$

where  $m$  is the number of constraints.

If only one constraint is not satisfied ( $f_i = 0$ ), the specific robot configuration is avoided, since  $f = 0$ .

A set of constraints which keeps the robot away from its joint limits, ensuring a natural configuration, is:

$$f_j^*(\theta_j) = \left( \sin \left( \frac{\theta_j - \theta_j^{\min}}{\theta_j^{\max} - \theta_j^{\min}} \cdot \pi \right) \right)^{\gamma_j} \quad (8)$$

where  $1 \leq j \leq 6$  is the joint number.

Static constraints can be represented graphically as gray level configuration maps. An example is given in Figure 8 and 9, where the two dimensions of the map are the rotary angle  $\theta_R$  and the discrete time  $t$ . The planner attempts to find a path from a start configuration (leftmost column in the map) to a final configuration (rightmost column in the map) which, while obeying all the constraints at least partly, has to be as smooth as possible and also have to obey the maximum angular speed and acceleration values for the rotary table. The robot motion is not constrained explicitly, but high speeds in robot motions can be avoided by adding static constraints which do not allow the robot to reach singular configurations in the kinematical chain.

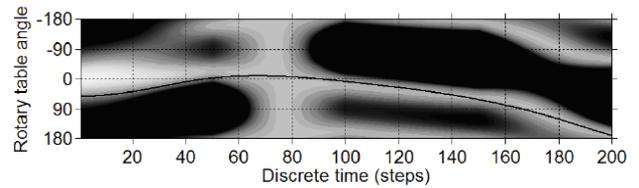


Figure 8 Grayscale configuration map for static constraints.

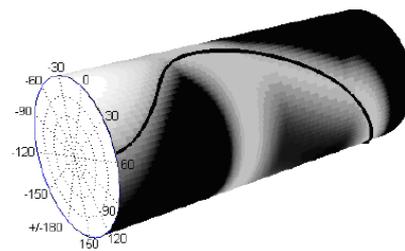


Figure 9 Cylindrical view of the configuration map.



Figure 10 Geometric models of the 3D scanning system components.



Figure 11 3D models for robot and laser probe: accurate (for rendering), convex and intermediate oriented boxes (for fast collision detection)

Dynamic constraints for the rotary table are specified using scalar weights for angular speed and acceleration,  $k_a$  and  $k_\omega$ . The heuristic algorithm, called *Ray Shooting*, attempts to try various constant-acceleration paths (rays) and selecting the one which has the lowest cost, at every time step. This strategy ensures low variations in the angular speed of the rotary table, which allows scanned parts to sit on the table without any additional fixture. The planning algorithm also attempts to reduce scanning time, energy consumption and increased scanning accuracy by avoiding near-singular configurations.

A constraint suitable for avoiding collision detection depends on the *minimal distance* between two rigid bodies,  $d_{min}$ . If  $d_{min}$  is less than a threshold  $d_{min}^{low}$ , the constraint is not satisfied, and this configuration is forbidden (the planner will never drive the robot through this configuration). If  $d_{min}$  is higher than  $d_{min}^{high}$ , the constraint is fully satisfied:

$$f_C(d_{min}) = \begin{cases} 0, d_{min} < d_{min}^{low} \\ \sin\left(\frac{d_{min} - d_{min}^{low}}{d_{min}^{high} - d_{min}^{low}} \cdot \pi\right)^{\gamma_C} \\ 1, d_{min} > d_{min}^{high} \end{cases} \quad (9)$$

The exponent  $\gamma_C$  controls the constraint intensity for  $d_{min}$  between  $[d_{min}^{low} \dots d_{min}^{high}]$ : higher values rejects values closer to  $d_{min}^{low}$ , while lower values are more permissive.

Of course, two successive bodies in the kinematical chain should not be tested for collisions, since they are always touching each other.

The parameters  $d_{min}^{low}$  and  $d_{min}^{high}$  can be chosen the same for every pair of possibly colliding bodies, or can be adjusted for each pair. For example, the distance between the laser sensor and 4<sup>th</sup> robot link is by design 10 millimeters, so there's no point in setting a higher value for  $d_{min}^{high}$ . However, if one has to keep the distance between the laser sensor and the rotary table at least 20 mm, and preferably 50 mm, then  $d_{min}^{low}$  and  $d_{min}^{high}$  should be set to 20 and 50 respectively.

## 5 IMPLEMENTATION ISSUES

The elements involved in collision detection are geometric models of their physical counterpart (Figure 10). For the robot arm, the meshes are imported from the CAD files. The rotary table has a very simple mechanical structure, and can be modeled as a stack of two cylinders. The laser sensor did not come with a 3D CAD file, but its shape was easily reconstructed from its dimensions, since it did not have a complex shape.

Other surrounding objects in the robot space can be digitized with the laser scanner itself. These objects include the CNC milling machine, the working table mounted near the robot arm and the mechanical parts placed on the table. State-of-art collision detection algorithms are able to exploit the convexity of the geometries in order to reduce computation time and provide smooth real-time operation. Figure 11 shows the difference between the complete 3D robot model (left) and the one composed from the convex hulls of each segment (right). The number of triangles in each complete (and concave) mesh is 5000; the convex hulls have between 350 and 1200 faces. From a visual point of view, the meshes are not suitable for rendering, but would give satisfactory results for collision detection even without the full meshes.

A protective convex hull, which completely encloses the underlying geometry and provides a safety margin around each link, was proposed in [9]. This also has the additional advantage that the triangle mesh can be simplified even more, for further speed increase. Convex hull and mesh simplification were performed with the open source tool MeshLab [10].

Fast implementations use preliminary tests with simple geometric primitives, e.g. bounding spheres (which are easiest to check) and bounding boxes (which may approximate the true shape better). Bounding boxes can be aligned with the World axes (AABB - Axis Aligned Bounding Box) or can have arbitrary orientation (OBB - Oriented Bounding Box), e.g. the one which minimizes the box volume. For articulated bodies, a hybrid approach can be used: the AABB is computed for each segment at the beginning of the simulation; then, while the articulated body moves, the AABBs are reoriented using the transformation matrix for each link (Figure 11, right). In this way, only the box corners have to be transformed during the articulated motion. For pure AABB and OBB models, the bounding box would have to be computed at every step from a much larger data set, which would limit the usefulness of the bounding boxes.

Bounding boxes alone are not suitable for collision detection, since they may give many false positives. However, it is also possible to approximate a shape using hierarchies of AABBs, OBBs or bounding spheres.

### 5.1 SOFTWARE IMPLEMENTATIONS OF COLLISION DETECTION

There are two main classes of publicly available libraries which implement collision detection:

- rigid body dynamics simulators, used for computer games;
- standalone libraries for collision and proximity queries.

### 5.1.1 Rigid body dynamics simulation packages

There are numerous rigid body dynamics simulators which implement state-of-art collision detection algorithms. They may be used for collision queries alone; however, with a possible computational overhead; if this represents an issue, a dedicated library should be used.

Rigid body engines available under proprietary licenses include NVidia Physx (formerly known as AGEIA and Novodex), Intel HAVOK, Newton Game Dynamics and True Axis. There are also engines available under public licenses, such as BSD, ZLib and GPL, including Open Dynamics Engine (ODE), Bullet, JigLib and Tokamak. Most of the above engines can be wrapped in a unified abstraction system, like PAL, OPAL and GangstaWrapper. A comparative test among engines supported by PAL may be found in [11].

### 5.1.2 Libraries for discrete collision queries

The following libraries are limited to convex polyhedra:

- GJK - Gilbert, Johnson and Keerthi distance routine; runs in expected constant time [12]; used in Bullet Physics;
- I-COLLIDE: exact collision detection for large environments [13]; uses Lin-Canny Closest Features, and is implemented in used in "Impulse" rigid body simulator [14];
- SWIFT: supports also bodies composed of convex pieces. Queries: clash, distance and contact determination [15];

The following libraries can handle non-convex polyhedra, also known as polygon soups:

- RAPID: uses OBBTree, a hierarchy of OBBs [16];
- PQP: intersection, distance and tolerance verification [17];
- V-COLLIDE: works with a large number of objects; uses 3 tests: n-body, hierarchical OBB and exact. [18].
- SWIFT++: for arbitrary polyhedral models. Queries: clash, tolerance, distance and contact determination [19].
- V-CLIP: Voronoi Clip algorithm; similar to Lin-Canny, with less complexity and improved robustness [20].
- OPCODE: memory-optimized AABB-tree [21]
- GIMPACT: implemented in ODE and Bullet simulators.

### 5.1.3 Libraries for continuous collision detection (CCD)

- FAST: CCD for general, rigid polyhedra [22];

- CATCH: CCD for articulated models [23]; uses FAST for solid body CCD and SWIFT++ for distance queries.

A proof-of-concept implementation of robot collision detection was implemented using Open Dynamics Engine as a wrapper for collision detection, which uses OPCODE and GIMPACT libraries. The simulation is based on the framework presented in [24], and it was implemented in Python, using PyODE for low-level ODE calls and cgkit (Python Computer Graphics Kit) for importing triangular meshes.

## 6 CONCLUSION

The paper presented practical considerations about implementing a predictive collision detection routine, which improves robustness of existing robotic applications. A watchdog task runs on a PC workstation, continuously monitoring the robot motion, while having also knowledge of geometries of robot arm and nearby equipment. The watchdog intercepts the robot motion without disturbing the application program, using a prediction scheme when necessary. Collision detection queries for the entire scene run in 10 milliseconds on a Core2Duo CPU, therefore the bottleneck is the communication layer between the PC and robot. Future work will improve robustness and generality for high-speed robot applications.

## ACKNOWLEDGMENT

This work was funded by the National Council for Scientific University Research, in the framework of the National Plan for Research, Development and Innovation, grant 69/2007.

## REFERENCES

- [1] Min Tang, Dinesh Manocha, Jiang Lin, Ruo-Feng Tong, Collision-Streams: Fast GPU-based Collision Detection for Deformable Models, *ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games (I3D)*, pp. 63-70, 2011.
- [2] Jia Pan, Dinesh Manocha, GPU-based Parallel Collision Detection for Real-time Motion Planning, *Algorithmic Foundations of Robotics IX: Selected Contributions of the Ninth International Workshop on the Algorithmic Foundations of Robotics (WAFR), Springer Tracts in Advanced Robotics (STAR)*, Vol. 68, pp. 211-228, 2011.
- [3] Harish Pungotra, George K. Knopf, and Roberto Canas, Efficient algorithm to detect collision between deformable B-spline surfaces for virtual sculpting, *Computer-Aided Design*, Vol. 40, Issues 10-11, 2008, Pages 1055-1066.
- [4] M. C. Lin and S. Gottschalk, Collision detection between geometric models: A survey. *Proc. of IMA*

- Conference on Mathematics of Surfaces*, 1998, pp. 37-56.
- [5] P. Jimnez, F. Thomas, and C. Torras, 3D collision detection: a survey. *Computers and Graphics*, Vol. 25, No. 2, pp. 269- 285, 2001.
- [6] A. Dumitrache, T. Borangiu, and A. Dogar, Automatic generation of milling toolpaths with tool engagement control for complex part geometry. *Proceedings of Intelligent Manufacturing Systems*, Lisbon, Portugal, 2010.
- [7] Adept Technology, Inc., *Adept SmartMotion Developer's Guide*. 2004.
- [8] T. Borangiu, A. Dogar, and A. Dumitrache, A heuristic approach for constrained real time motion planning of a redundant 7-dof mechanism for 3D laser scanning. *INCOM 2009 - 13th IFAC Symposium on Information Control Problems in Manufacturing*, Moscow, Russia.
- [9] J. Kuffner, K. Nishiwaki, S. Kagami, K. N. S. Kagami, Y. Kuniyoshi, M. Inaba, and H. Inoue, Self-collision detection and prevention for humanoid robots. *Proceedings of the IEEE International Conference on Robotics and Automation*, 2002, pp. 2265-2270.
- [10] P. Cignoni, M. Corsini, and G. Ranzuglia, Meshlab: an open-source 3D mesh processing system. *ERCIM News*, No. 73, pp. 45-46, April 2008.
- [11] A. Boeing and T. Braunl, Evaluation of real-time physics simulation systems. *GRAPHITE '07: Proceedings of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia*. New York, NY, USA: ACM, 2007, pp. 281-288.
- [12] G. Van den Bergen, A fast and robust GJK implementation for collision detection of convex objects. *J. Graph. Tools*, Vol. 4, No. 2, pp. 7-25, 1999.
- [13] J. D. Cohen, M. C. Lin, D. Manocha, and M. Ponamgi, I-Collide: an interactive and exact collision detection system for large-scale environments. *I3D '95: Proceedings of the 1995 symposium on Interactive 3D graphics*, New York, NY, USA: ACM, 1995, pp. 189-ff.
- [14] B. V. Mirtich, *Impulse-based dynamic simulation of rigid body systems*, Ph.D. dissertation, 1996.
- [15] S. A. Ehmman and M. C. Lin, Accelerated proximity queries between convex polyhedra by multi-level voronoi marching. *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2000, pp. 2101-2106.
- [16] S. Gottschalk, M. C. Lin, and D. Manocha, Obbtrees: a hierarchical structure for rapid interference detection. *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, New York, NY, USA: ACM, 1996, pp. 171-180.
- [17] E. Larsen, S. Gottschalk, M. C. Lin, and D. Manocha, *Fast proximity queries with swept sphere volumes*. Tech. Rep., 1999.
- [18] T. C. Hudson, M. C. Lin, J. Cohen, S. Gottschalk, and D. Manocha, V-collide: accelerated collision detection for vrml. *VRML '97: Proceedings of the second symposium on Virtual reality modeling language*, New York, NY, USA: ACM, 1997, pp. 117-ff.
- [19] S. A. Ehmman and M. C. Lin, Accurate and fast proximity queries between polyhedra using convex surface decomposition. *Computer Graphics Forum*, 2001, pp. 500-510.
- [20] B. Mirtich, V-clip: fast and robust polyhedral collision detection. *ACM Trans. Graph.*, Vol. 17, No. 3, pp. 177-208, 1998.
- [21] P. Terdiman, *Memory-optimized bounding-volume hierarchies*. March 2001.
- [22] X. Zhang, M. Lee, and Y. J. Kim, Interactive continuous collision detection for non-convex polyhedra. *Vis. Comput.*, Vol. 22, No. 9, pp. 749-760, 2006.
- [23] X. Zhang, S. Redon, M. Lee, and Y. J. Kim, Continuous collision detection for articulated models using taylor models and temporal culling. *Proceedings of SIGGRAPH 2007*, Vol. 26, No. 3, p. 15, 2007.
- [24] T. Borangiu, A. Dogar, and A. Dumitrache, Modelling and simulation of short range 3D triangulation-based laser scanning system. *Int. Journal of Computers, Communications and Control (IJCCC)*, Vol. 3, Suppl. Issue - ICCCC'08, pp. 190-195, 2008.

# TEMPLATE FOR PREPARING PAPERS FOR PUBLISHING IN INTERNATIONAL JOURNAL OF MECHANICS AND CONTROL

Author1\*      Author2\*\*

\* affiliation Author1

\*\* affiliation Author2

## ABSTRACT

This is a brief guide to prepare papers in a better style for publishing in International Journal of Mechanics and Control (JoMaC). It gives details of the preferred style in a template format to ease paper presentation. The abstract must be able to indicate the principal authors' contribution to the argument containing the chosen method and the obtained results. (max 200 words)

Keywords: keywords list (max 5 words)

## 1 TITLE OF SECTION (E.G. INTRODUCTION)

This sample article is to show you how to prepare papers in a standard style for publishing in International Journal of Mechanics and Control.

It offers you a template for paper layout, and describes points you should notice before you submit your papers.

## 2 PREPARATION OF PAPERS

### 2.1 SUBMISSION OF PAPERS

The papers should be submitted in the form of an electronic document, either in Microsoft Word format (Word'97 version or earlier).

In addition to the electronic version a hardcopy of the complete paper including diagrams with annotations must be supplied. The final format of the papers will be A4 page size with a two column layout. The text will be Times New Roman font size 10.

## 2.2 DETAILS OF PAPER LAYOUT

### 2.2.1 Style of Writing

The language is English and with UK/European spelling. The papers should be written in the third person. Related work conducted elsewhere may be criticised but not the individuals conducting the work. The paper should be comprehensible both to specialists in the appropriate field and to those with a general understanding of the subject. Company names or advertising, direct or indirect, is not permitted and product names will only be included at the discretion of the editor. Abbreviations should be spelt out in full the first time they appear and their abbreviated form included in brackets immediately after. Words used in a special context should appear in inverted single quotation mark the first time they appear. Papers are accepted also on the basis that they may be edited for style and language.

### 2.2.2 Paper length

Paper length is free, but should normally not exceed 10000 words and twenty illustrations.

### 2.2.3 Diagrams and figures

Figures and Tables will either be entered in one column or two columns and should be 80 mm or 160 mm wide respectively. A minimum line width of 1 point is required at actual size. Captions and annotations should be in 10 point with the first letter only capitalised *at actual size* (see Figure 1 and Table VII).

---

Contact author: author1<sup>1</sup>, author2<sup>2</sup>

<sup>1</sup>Address of author1.

<sup>2</sup>Address of author2 if different from author1's address  
E-mail: author1@univ1.com , author2@univ2.com

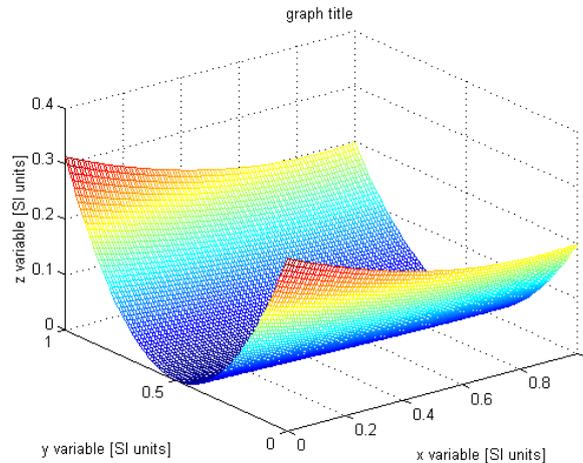


Figure 1 Simple chart.

Table VII - Experimental values

Robot Arm Velocity (rad/s)	Motor Torque (Nm)
0.123	10.123
1.456	20.234
2.789	30.345
3.012	40.456

#### 2.2.4 Photographs and illustrations

Authors could wish to publish in full colour photographs and illustrations. Photographs and illustrations should be included in the electronic document and a copy of their original sent. Illustrations in full colour ...

#### 2.2.5 Equations

Each equation should occur on a new line with uniform spacing from adjacent text as indicated in this template. The equations, where they are referred to in the text, should be numbered sequentially and their identifier enclosed in parenthesis, right justified. The symbols, where referred to in the text, should be italicised.

- point 1
  - point 2
    - point 3
- 1. numbered point 1
- 2. numbered point 2
- 3. numbered point 3

$$W(d) = G(A_0, \sigma, d) = \frac{1}{T} \int_0^{+\infty} A_0 \cdot e^{-\frac{d^2}{2\sigma^2} t} dt \quad (1)$$

### 3 COPYRIGHT

Authors will be asked to sign a copyright transfer form prior to JoMaC publishing of their paper. Reproduction of any part of the publication is not allowed elsewhere without permission from JoMaC whose prior publication must be cited. The understanding is that they have been neither previously published nor submitted concurrently to any other publisher.

### 4 PEER REVIEW

Papers for publication in JoMaC will first undergo review by anonymous, impartial specialists in the appropriate field. Based on the comments of the referees the Editor will decide on acceptance, revision or rejection. The authors will be provided with copies of the reviewers' remarks to aid in revision and improvement where appropriate.

### 5 REFERENCES (DESCRIPTION)

The papers in the reference list must be cited in the text. In the text the citation should appear in square brackets [ ], as in, for example, "the red fox has been shown to jump the black cat [3] but not when...". In the Reference list the font should be Times New Roman with 10 point size. Author's first names should be terminated by a 'full stop'. The reference number should be enclosed in brackets. The book titles should be in *italics*, followed by a 'full stop'. Proceedings or journal titles should be in *italics*. For instance:

#### REFERENCES (EXAMPLE)

- [1] Smith J., Jones A.B. and Brown J., *The title of the book*. 1st edition, Publisher, 2001.
- [2] Smith J., Jones A.B. and Brown J., The title of the paper. *Proc. of Conference Name*, where it took place, Vol. 1, paper number, pp. 1-11, 2001.
- [3] Smith J., Jones A.B. and Brown J., The title of the paper. *Journal Name*, Vol. 1, No. 1, pp. 1-11, 2001.
- [4] Smith J., Jones A.B. and Brown J., *Patent title*, U.S. Patent number, 2001.

*International Journal of Mechanics and Control – JoMaC*  
Published by Levrotto&Bella  
**TRANSFER OF COPYRIGHT AGREEMENT**

<p>NOTE: Authors/copyright holders are asked to complete this form signing section A, B or C and mail it to the editor office with the manuscript or as soon afterwards as possible.</p>	<p><i>Editor's office address:</i> Andrea Manuello Bertetto Elvio Bonisoli <i>Dept. of Mechanics</i> <i>Technical University – Politecnico di Torino</i> <i>C.so Duca degli Abruzzi, 24 – 10129 Torino – Italy</i> <i>e_mail: jomac@polito.it</i> <i>fax n.: +39.011.564.6999</i></p>
--	---

The article title:

---

By: \_\_\_\_\_

To be Published in *International Journal of Mechanics and Control JoMaC*  
*Official legal Turin court registration Number 5320 (5 May 2000) - reg. Tribunale di Torino N. 5390 del 5 maggio 2000*

A Copyright to the above article is hereby transferred to the JoMaC, effective upon acceptance for publication. However the following rights are reserved by the author(s)/copyright holder(s):

1. All proprietary rights other than copyright, such as patent rights;
2. The right to use, free or charge, all or part of this article in future works of their own, such as books and lectures;
3. The right to reproduce the article for their own purposes provided the copies are not offered for sale.

*To be signed below by all authors or, if signed by only one author on behalf of all co-authors, the statement A2 below must be signed.*

A1. All authors:

SIGNATURE \_\_\_\_\_ DATE \_\_\_\_\_ SIGNATURE \_\_\_\_\_ DATE \_\_\_\_\_

PRINTED NAME \_\_\_\_\_ PRINTED NAME \_\_\_\_\_

SIGNATURE \_\_\_\_\_ DATE \_\_\_\_\_ SIGNATURE \_\_\_\_\_ DATE \_\_\_\_\_

PRINTED NAME \_\_\_\_\_ PRINTED NAME \_\_\_\_\_

A2. One author on behalf of all co-authors:

*"I represent and warrant that I am authorised to execute this transfer of copyright on behalf of all the authors of the article referred to above"*

PRINTED NAME \_\_\_\_\_

SIGNATURE \_\_\_\_\_ TITLE \_\_\_\_\_ DATE \_\_\_\_\_

B. The above article was written as part of duties as an employee or otherwise as a work made for hire. As an authorised representative of the employer or other proprietor. I hereby transfer copyright to the above article to *International Journal of Mechanics and Control* effective upon publication. However, the following rights are reserved:

1. All proprietary rights other than copyright, such as patent rights;
2. The right to use, free or charge, all or part of this article in future works of their own, such as books and lectures;
3. The right to reproduce the article for their own purposes provided the copies are not offered for sale.

PRINTED NAME \_\_\_\_\_

SIGNATURE \_\_\_\_\_ TITLE \_\_\_\_\_ DATE \_\_\_\_\_

C. I certify that the above article has been written in the course of employment by the United States Government so that no copyright exists, or by the United Kingdom Government (Crown Copyright), thus there is no transfer of copyright.

PRINTED NAME \_\_\_\_\_

SIGNATURE \_\_\_\_\_ TITLE \_\_\_\_\_ DATE \_\_\_\_\_

## CONTENTS

- 3 The Development of a MEMS/NEMS-Based 3 D.O.F. Compliant Micro Robot**  
*M. Balucani, N.P. Belfiore, R. Crescenzi, M. Verotti*
- 11 Some Issues on Holonic Systems Analysis, Design and Implementation**  
*D. Panescu, C. Pascal*
- 19 Optimizing Parameters of Trajectory Representation for Movement Generalization: Robotic Throwing**  
*A. Gams, T. Petrič, L. Žlajpah, A. Ude*
- 27 Pneumatronic Unit for Motion of Bows**  
*E. Ravina*
- 35 A Multi-Behaviour Algorithm for Auto-Guided Movements in Surgeon Assistance**  
*E. Bauzano, V.F. Muñoz, I. Garcia-Morales*
- 43 Quadrotor Control Based on Partial Sensor Data**  
*L. Kis, B. Lantos*
- 51 Simulation Investigations on a Novel Approach to Model Reference Adaptive Controllers**  
*J.K. Tar, I.J. Rudas, J.F. Bitó, K.R. Kozłowski, C. Pozna*
- 61 Time-Optimal Motion Planning for Robots**  
*S. János, M. József*
- 75 Isotropy in Any RR Planar Dyad under Active Joint Stiffness Regulation**  
*N.P. Belfiore, P. Di Giamberardino, I.J. Rudas, M. Verotti*
- 83 CAD Based Techniques for Workspace Analysis and Representation of the 3-CRS Parallel Manipulator**  
*K. Assad Arrouk, B.C. Bouzgarrou, G. Gogu*
- 91 An Experimental Evaluation of Earthquake Effects on Mechanism Operation**  
*S. Sula, M. Ceccarelli, D. Pisla*
- 99 Exercise Device for Upper-Extremity Sensory-Motor Capability Augmentation Based on Magneto-Rheological Fluid Actuator**  
*R. Kamnik, J. Perdan, T. Bajd, M. Munih*
- 105 Laboratory Test for an Integrated Device in Agricultural Applications**  
*A. Manuello Bertetto, C. Falchi, R. Pinna, R. Ricciu*
- 113 Autonomous Sensor Networks for Monitoring Industrial Processes with Application to Mobile Robots**  
*G. Balakrishnan, S.S. Hiremath*
- 125 Collision and Proximity Avoidance for Robust Behaviour of Real-Time Robot Applications**  
*A. Dumitrache, T. Borangiu*