VIRTUAL TO REAL IN ROBOTIC ASSEMBLY/DISASSEMBLY TASKS

Diana POPESCU[1], Robert IACOB[2], Radu PARPALA[3], Tiberiu DOBRESCU[4]

Currently, the focus of research in the field of Design for Assembly/Disassembly (DfA/D) is put on the automatic generation of A/D sequences. This represents a key aspect because having, from the design phase, the information on the valid A/D components' movements can be useful in the early determination of the structure, specifications and working program of a device, robot or tool adapted to perform the operations in question. In this sense, the paper presents the integration between a software tool for automatic generation of A/D sequences and an industrial robot programmed to directly use these data for its motion control.

Keywords: assembly/disassembly, virtual simulation, industrial robot, interface, mobility, sequences

1. Introduction

Today's producers, in the context of a highly dynamic global market and very tough industrial competition, need to find modalities to shorten the time-to-market, to design and manufacture fast, high quality and cost effective products, before competition, in the same time fulfilling the expectations of more and more demanding consumers – looking for products of good quality, customizable and cheap –, respecting legal regulations and constraints, and obtaining profit [1]. Some of these goals can be achieved throughout a permanent focus on innovating [2], optimizing and integrating different aspects and phases involved in the Product Lifecycle (PLC). Therefore, researches are continuously made to develop proper and efficient software tools for supporting engineers' work in all PLC stages [3], A/D modeling and simulation representing a significant part of this demarche.

[1] Associate Professor, Department Machine and Production Systems, University POLITEHNICA of Bucharest, Romania, e-mail: diana@mix.mmi.pub.ro

[2] Lecturer, Department Machine and Production Systems, University POLITEHNICA of Bucharest, Romania, e-mail: robert.iacob@gmail.com

[3] Lecturer, Department Machine and Production Systems, University POLITEHNICA of Bucharest, Romania, e-mail: radu.parpala@gmail.com

[4] Professor, Department Machine and Production Systems, University POLITEHNICA of Bucharest, Romania, e-mail: tibidobrescu@yahoo.com

In this context, the current paper is considering the presentation of a software tool called ViPAD (Virtual Platform for A/D simulation) for the assessment of design changes' impact on the assembly, repair and maintenance times and costs, on the recycling or on components re-using operations, but as well on the robots mounting or dismounting tasks programming. The developed application allows the generation of valid A/D trajectories (translations, rotations and helical ones) of a component from an assembly with respect to its neighbors, starting from the 3D CAD (Computer-Aided Design) model of the product. It offers the possibility to automatically identify contacts from a mechanical assembly, to calculate the mobility of a component and to determine the optimal A/D sequence relative to user's criteria.

The core of this research is the robot programming module that uses different types of data for programming the motions of a robot integrated in mounting or dismounting cells (applications), as will be presented further, contributing to the decrease of the time and cost associated to these tasks. Thus, the virtual A/D sequences generated by the software are directly transformed into teaching points defining the robot trajectories, strictly related to the components' insertion/extraction into/from the assembly, without the need of additional software or data processing. To these trajectories, in some cases, supplementary data should be added from the virtual 3D model of the robotic cell in order to define some particular points of interest, e.g. storage areas, conveyors etc.

2. Robotic programming based on CAD data

Computer-Aided Robotic (CAR) general applications such as Delmia (Dassault Systemes) or RobCAD (Technomatix), as well as robot producers' software like RobotStudio (ABB Robotics) or KukaSim (Kuka Robotics) are currently used for the Off-Line Programming (OLP) of robots integrated in different types of applications. They offer software tools for generating and simulating robots' kinematics, enabling engineers to design and virtually test different robotic cells configurations and, thus, to optimize robot's trajectories. A typical application of these software packages consists in bringing together detailed 3D CAD models for all components (robots, conveyors, machine-tools, equipment, etc.) and for the work-piece, placing them in a certain configuration/architecture on the work floor and then simulating and optimizing the entire cell functioning while running the robots' predefined program. It is important to note that, despite the advantages, the cost of these applications is high and their use requires high technical specialization.

Another classical programming method is based on robot teach pendant, which is a difficult and time-consuming task and often prone to inaccuracies. It also requires user's technical expertise for a specific type of robot/specific robot programming language.

Pan et al. [4], in their review on the advances in robots' programming, mention as new programming solutions the use of Augmented Reality (AR), Virtual Reality (VR) or Human Computer Interfaces (HMI) and gesture recognitions, which provide a more intuitive perspective over OLP. As an example, an OLP approach based on VR, presented in detail by Jen et al. [5], consists in using a Virtual Environment (VE), data glove and sensor trackers for teaching (or OLP) a KUKA robot to perform a palletizing operation (Fig. 1). The users' hand movements in the VE are tracked and saved in a database, as many attempts are necessary can be performed for optimizing the robot trajectory. This trajectory is then used for robot programming, in a Point-to-Point (PtP) linear motion approach. It should be noted that the approach is semi-automated.

Also, Chong et al. describe in [6] an OLP application based on AR for planning a collision-free robot trajectory, with the advantages of an intuitive programming interface and the possibility to avoid building virtual models for each component of a robotic cell (Fig. 2).

These are only two examples of applications based on advanced techniques which can make the user interaction with the robot and programming tasks easier. However, they require the use of VR specific equipment and modeling these VE for simulations necessitates time and highly specific knowledge. Therefore, we consider that it is important to develop solutions which can automatically extract data from the 3D CAD files and use it as input for generating robots' trajectories.

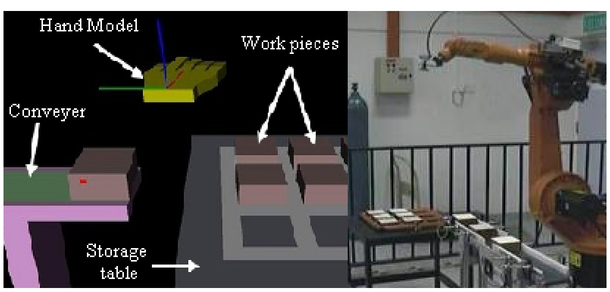


Fig. 1 Virtual Teaching System application for OLP of a KUKA robot [5]

Fig. 2 Generating a robot trajectory in an AR [6]

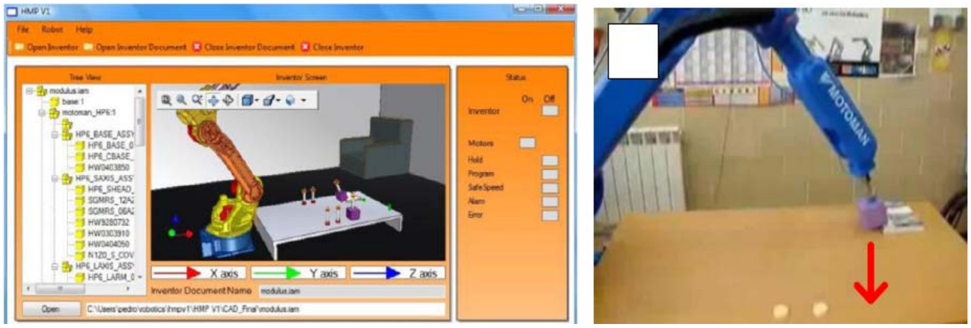


Fig. 3 OLP platform for extracting robot paths from the CAD drawing of the cell [10]

Thus, neither expensive devices nor CAR software are necessary, in the same time maintaining a simple and clear modality to program the robot by transferring it a list of points and instructions that define its working trajectory for a certain application.

In this category enrolls the applications of Kim [7], Nagata [8], Chen [9], Neto [10], all being CAD-based offline robot programming for pick-and-place, spray painting or applying adhesives applications. Although the idea to use CAD data for robot programming is under research since the '80, to the best of our knowledge, no research has been done for programming a robot based on valid A/D sequences automatically generated from the 3D CAD data of a product assembly, as is the case of the method and application proposed in this research.

The application presented by Nagata [8] is using data generated by a 3D CAD/CAM system and directly transferred to the robot as tool (cutter) locations for an application of free-form surfaces sanding. Also, information on polishing force, sanding motor power and feed rate are added to the cutting location data. With this approach, the operator's time consuming task to input a large number of points in order to teach the robot to follow a complicated path is avoided and, also, accuracy is ensured.

A newer approach, closer to our research perspective, is described in [10]. Using Autodesk Inventor API, starting with the CAD drawing of a robotic cell, data are automatically extracted, i.e. points corresponding to robot end-effector positions (materialized as a 3D trajectory model included in the cell and saved apart) on each segment of its path, relative to the work coordinate system (Fig. 3).

The robotic cell should not be precisely modeled in terms of component appearance, thus saving important time and effort. The user is planning the robot path in the CAD model of the cell, a novel OLP platform being used for extracting data, mapping them on the real environment and calibrating tasks. The developed platform main purpose is to integrate the robotic cell design, product design and robot programming. The authors mention that the proposed solution needs more improvements (kinematic singularities, joints limitation, or collisions not being yet deal with – a virtual simulation tool being useful for these cases) and it should not be consider a general solution, but a step forward CAD-based OLP. More, it should be noted that the robot trajectories are manually defined by the user in the design stage, which represent a major drawback.

In the coming years, different projects will be launched related to the integration of the latest industrial automation systems for assembly operations in combination with human capabilities. Focus is given to combining robot strength, velocity, predictability, repeatability and precision with human intelligence and skills to get at a hybrid solution that would be involving the safe cooperation of operators with autonomous and adapting robotic systems [11].

3. Software application for A/D sequences' generation and simulation

ViPAD is a software application integrated with Solid Works (SW) system, which can support the automatic generation of A/D sequences based on the 3D CAD model of the product assembly [12]. This is possible because ViPAD:

□ integrates, in a set of interfaces for a product, all the information on the geometrical constraints, relative position of contact surfaces, contact common area, information about the neighboring components and their relative mobility;

□ includes a general mobility operator which is used for evaluating the possible movements of the assemblies' components and thus for generating the valid trajectories of a component relative to its neighbors, and for the real-time graphical representation of the components mobilities;

□ computes mechanical joints which along with a set of predefined rules for different types of standard components (e.g. rivets, pegs, screws, etc.), allowing to find the functional components of an assembly.

Figure 4 represents ViPAD general architecture. Modules 1 to 6 are briefly presented below, being already detailed in [12], while the Robot programming module (M.7) will be presented in the next section. The modules are integrated in an innovative environment providing to the design and automation engineers the necessary tools for optimizing the A/D modeling processes, offering useful information for the whole PLC, from design and fabrication to recycling.

From a user point of view, the first five modules represent the invisible part of the application that computes all the necessary information for M.6 and M.7:

M.1. Import & data representation module – the 3D CAD model of the assembly product in STEP format is imported in the application, a new data structure being developed for handling different types of representations.

M.2. Interfaces module – allows the identification of: geometric constraints, contact surfaces relative position, common area; this information is added to the product data structure and further used in other analysis modules.

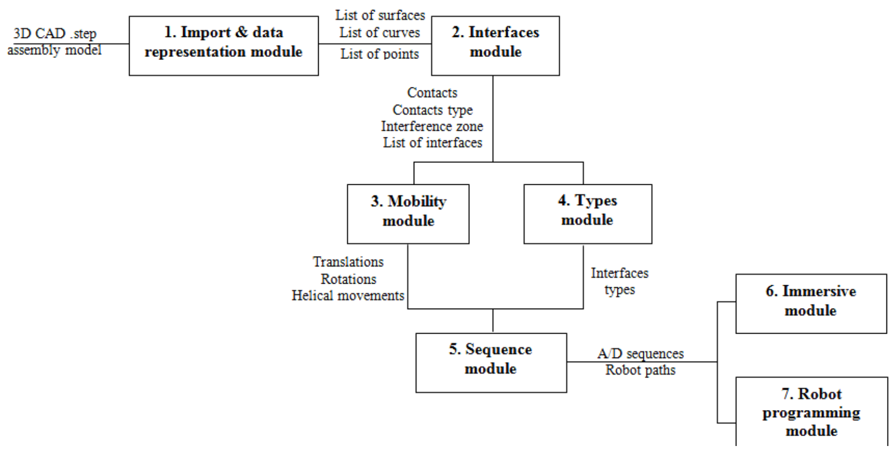


Fig. 4 ViPAD architecture and data flow

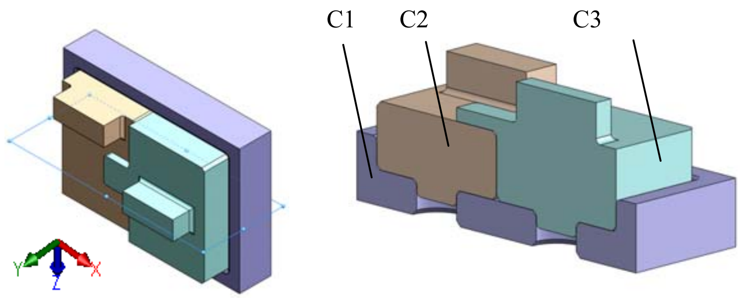
M.3. Mobility module – is developed to generate all the valid A/D trajectories and it is based on an innovative method and operator which can compute and represent translations, rotations, helical movements for a component of the assembly or for the whole assembly.
M.4. Types module – is used to find assembly components with a specific functional role, based on the information generated by the interface module and a set of rules, defined for different types of components (e.g. screw, washer etc.).
M.5. Sequence module – determines the A/D sequences for an assembly component or for the whole product; all feasible sequences are generated and the best one (or the near-optimal one) can be identified according to pre-defined user's criteria.

The Immersive module (M.6) was the first fully visible module. It was developed as connected to CVE (Collaborative Virtual Environment) application and allows real-time simulation of A/D operations [13].

4. Robot programming module

The Robot Programming Module (RPM) is an interface between ViPAD and robot programming language. It is used for directly transfer A/D sequences, expressed as a standard text file, to the robot. Additional information regarding robot end-effector open/close positions, tools orientations in respect to the global coordinated system of the cell (available in SolidWorks), accurate positions of the cell's components etc., are also added to this file for a complete metadata export.

A product composed of 3 components denoted: C1, C2, C3 (Fig. 5) was chosen for A/D tests in the virtual robotic cell presented in figure 6. It should be noted that it is only a proof-of-concept example, the main purpose being to demonstrate the feasibility of the proposed method for real industrial applications.



a. b.

Fig. 5 Product-example: a. Isometric view; b. Section view

The virtual A/D cell (Fig. 6), designed for test purposes, integrates an ABB IRB-140 robot – programmed using the RAPID language, with a parallel finger gripper Schunk PGN-100, a robot controller, a belt conveyor and an A/D station. As a standard for this cell configuration, the robot calibration is made relative to the origin point of the assembly.

A typical working scenario with ViPAD–RPM is the following. SolidWorks software is used for detailed modeling the product. In addition, the main elements of the cell (not detailed virtual models) should be positioned, paying attention to the relative distances, in a different 3D layout for setting valid robot paths by reaching the desired positions and avoiding collisions. Then, the product assembly saved as STEP file is loaded in ViPAD and the A/D sequences are generated (Fig. 7) using the M1-M5. These modules are offering information not only about the order in which components should be moved into/from the assembly, but also about the movements on X, Y, Z axis (Fig. 8). Before launching RPM, M.6 module could be used by an operator for simulating the A/D tasks, thus checking the validity of the generated robot paths.

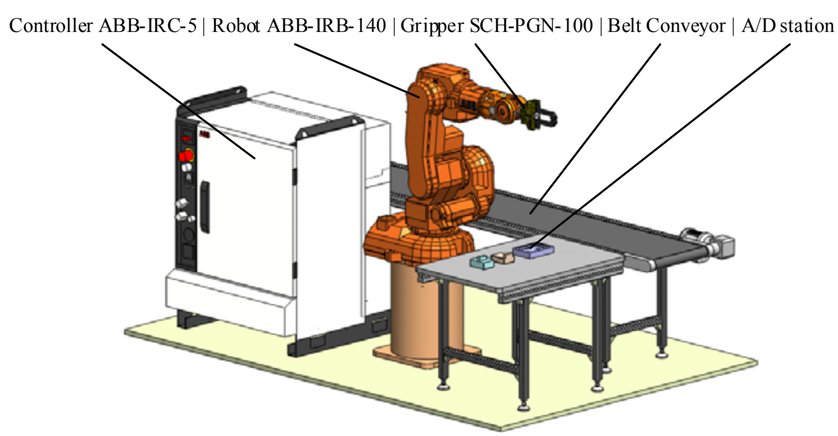


Fig. 6 Virtual robotic A/D cell

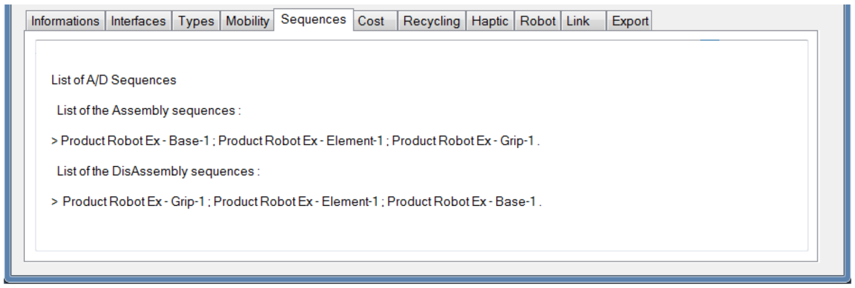


Fig. 7 A/D sequences generated by ViPAD

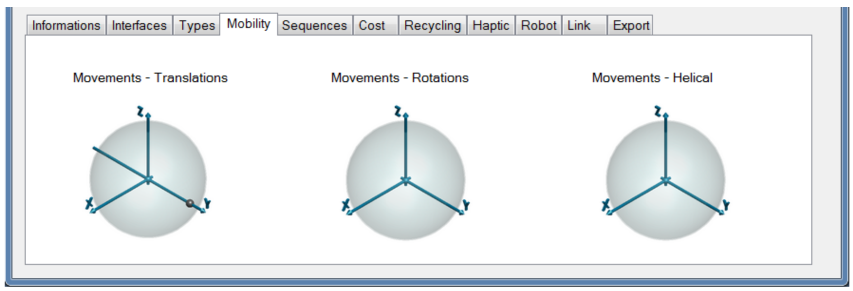


Fig. 8 Component C3 mobility computed by ViPAD

A text file containing the robot program is automatically generated by ViPAD. The program is based on the A/D sequence previously generated and, besides the robot functions, it contains the list of coordinates for the points representing components successive positions during A/D operations. These points are expressed relative to the Global Coordinated System (GCS) of the robotic cell, as established in SolidWorks. Each component is removed from the product assembly following the path indicated by ViPAD application, and brought to a predefined position, outside of the assembly (e.g. a conveyor for dismounted parts). From that point, the successive position of the end-effector is established considering the cell 3D representation.

All these points define the robot path, linear interpolation being applied between them. In order to accomplish its task, the robot end-effector should move on the trajectory resulted from ViPAD analysis, starting from the first point of the A/D path to its last point. When the robot reaches this last point of the A/D trajectory, the 3D CAD model of the whole cell is used for setting the rest of the points. Therefore, the control points which define the robot trajectory are obtained by concatenating these two lists of points. Using the inverse kinematics, the joints parameters are calculated. All the control points should be accessible by the end-effector, in case of joint singularities, the positions of the control points from the second list are change by considering predefined points' neighborhoods, so that the trajectory to remain collision free.

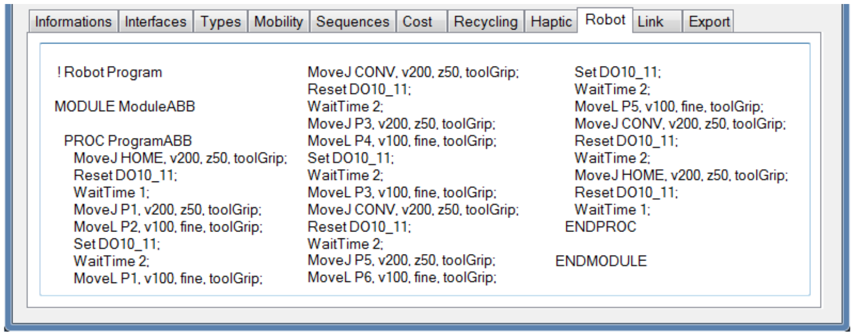


Fig. 9 ABB robot program generated through the ViPAD RPM module

In order to demonstrate the functioning of the programming module, the disassembly process of the product-example (Fig. 5) was performed in ViPAD and the generated robot program is shown in figure 9. All the three parts are dismounted and placed on conveyor. For this, two main functions are deployed: MoveJ – moves the robot quickly from one point to another by joint movement, and MoveL – moves the robot linearly. The points are computed by the proposed algorithm. Thus, for each part, two points are defined: gripping point – placed on the part, and the disassembly point – where the part is completely removed from the assembly (out of the bounding box).

5. Conclusions and further research

An automatic approach for robot program generation is presented in this paper for mechanical product mounting and dismounting. The robot control commands are generated and transmitted from the developed application to the real robot through a text file. This process uses different type of data computed from the product 3D model. In addition, some information related to the positioning of the cell elements is automatically retrieved or, in very particular cases, manually inserted.

In comparison to other CAD-based robot programming approaches, our research is focused on the A/D process. An application, developed for the automated generation of A/D sequences data of a mechanical assembly – namely ViPAD, is used for transferring different types of data to the robot, thus eliminating the need of programming using a teach pendant. This application represents a step forward because the movement' sequences are automatically exported from the software to the robot, thus exponentially reducing the time required to prepare an assembly or disassembly process.

Future work includes, in the first phase, different tests with a real ABB robotic A/D cell and, in a second phase, the implementation of more complex algorithms for a fully automated robot programming process.

Acknowledgment

The work has been partially funded by the EU-RO InnoRESEARCH project under Contract POSDRU/159/1.5/S/132395.

R E F E R E N C E S

[1]. *James Manyika, Jeff Sinclair, Richard Dobbs, Gernot Strube, Louis Rassey, Jan Mischke, Jaana Remes, Charles Roxburgh, Katy George, David O'Halloran, Sreenivas Ramaswamy* – "Manufacturing the future. The next era of global growth and innovation", McKinsey Global Institute Report, November 2012

[2]. *Hamzeh K. Bani Milhim, Xiaoguang Deng, Andrea Schiffauerova, Yong Zeng* – "The role of Product Lifecycle Management systems in organizational innovation", Product Lifecycle Management. Towards knowledge-rich enterprises vol. 388 of FIP Advances in Information and Communication Technology, pp. 322-331, 2012

[3]. *Tetsuo Tomiyama, Bart R. Meijer* – "Directions of next generation product development", in Advances in Design, pp.27-35, 2006

[4]. *Zengxi Pan, Joseph Polden, Nathan Larkin, Stephen van Duin, John Norrish* - "Recent progress on programming methods for industrial robots", in Robotics and Computer Integrated Manufacturing vol. 28, no. 2, pp. 87-94, 2012

[5]. *Yap Hwa Jen, Zahari Taha, Lee Jer Vui* - "VR-based robot programming and simulation system for an industrial robot", in International Journal of Industrial Engineering vol. 15, no. 3, pp. 314-322, 2008

[6]. *Jonathan Wun Shiung Chong, Patricia Soh-Khim Ong, Andrew Y.C. Nee, Kamal Youcef-Toumi* - "Robot programming using augmented reality: An interactive method for planning collision-free paths", in Robotics and Computer-Integrated Manufacturing vol. 25, no. 3, pp. 689-701, 2009

[7]. *J.Y. Kim* - "CAD-based automated robot programming in adhesive spray systems for shoe outsoles and uppers", in Journal of the Robotic Systems vol. 21, no. 11, pp. 625-634, 2004

[8]. *Fusaomi Nagata, Yukihiro Kusumoto, Yoshihiro Fujimoto, Keigo Watanabe* - "Robotic sanding system for new designed furniture with free-formed surface", in Robotics and Computer-Integrated Manufacturing vol. 23, no. 4, pp. 371-379, 2007

[9]. *Heping Chen, Weihua Sheng* - "Transformative CAD based industrial robot program generation", in Robotics and Computer-Integrated Manufacturing vol. 27, no. 5, pp. 942-948, 2011

[10]. *Pedro Neto, Nuno Mendes* - "Direct off-line robot programming via a common CAD package", in Robotics and Autonomous Systems vol. 61, pp. 896-910, 2013

[11]. *George Michalos, Sotiris Makris, Jason Spiliotopoulos, Ioannis Misios, Panagiota Tsarouchi, George Chryssolouris* - "ROBO-PARTNER: Seamless human-robot cooperation for intelligent, flexible and safe operations in the assembly factories of the future", in Procedia CIRP vol. 23, pp. 71-76, 2014

[12]. *Robert Iacob, Diana Popescu* - "Disassembly directions generation based on component mobility", in Studies in Informatics and Control journal vol. 22, no. 4, pp. 307-318, 2013

[13]. *Robert Iacob, Diana Popescu, Frederic Noel, Thibault Louis, Cedric Masclet, Patrick Maigrot* - "Haptic devices evaluation for industrial use", in Conference of the European Association of Virtual and Augmented Reality, pp. 39-45, 2014