Active Reconfiguration of Software and Hardware in a Robotic Workcell

Robert Bevec
Department of Automatics, Biocybernetics, and Robotics
Jožef Stefan Institute, Ljubljana, Slovenia
robert.bevec@ijs.si

Timotej Gašpar
Department of Automatics, Biocybernetics, and Robotics
Jožef Stefan Institute, Ljubljana, Slovenia
timotej.gaspar@ijs.si

Barry Ridge
Department of Automatics, Biocybernetics, and Robotics
Jožef Stefan Institute, Ljubljana, Slovenia
barry.ridge@ijs.si

Martin Bem
Department of Automatics, Biocybernetics, and Robotics
Jožef Stefan Institute, Ljubljana, Slovenia
martin.bem@ijs.si

Igor Kovač
Department of Automatics, Biocybernetics, and Robotics
Jožef Stefan Institute, Ljubljana, Slovenia
igor.kovac@ijs.si

Aleš Ude
Department of Automatics, Biocybernetics, and Robotics
Jožef Stefan Institute, Ljubljana, Slovenia
ales.ude@ijs.si

ABSTRACT
High volume production has been a prerequisite in order to invest into automation of the manufacturing process for decades. The high cost of setup and the inflexibility of the solution meant low batch productions, often present in *small and medium-sized enterprises (SMEs)*, were dismissed for automation. In order to bring automation closer to SMEs a flexible solution is required that can accommodate more than one manufacturing process, allows rapid change between them and doesn't require expertise knowledge for set up. In this paper we present a novel robotic workcell that enables active reconfiguration of software and hardware components, facilitating set up and production of several manufacturing processes on a single robotic cell. The ROS based software has been designed to be robot independent and modular. Special user interfaces have been developed for cell calibration, programming by demonstration and set up of quality control and part localization tasks. The proposed workcell is applicable in companies with product families, where manufacturing processes are similar and where fast changeover is required in order to adapt to new production requirements quickly. Due to the emphasis put on ease of use it will also be of interest to companies getting into automation for the first time.

1. INTRODUCTION
Fast changes in market demands lead to great fluctuation of orders down the manufacturing chain. Companies must react quickly, efficiently, and in an economically viable way. Robots have been successful in industrial production processes, when applied to repetitive tasks with long production runs and high unit volume. However, frequent shifts in the required product type or in the number of required products has prevented many companies from automating their manufacturing processes.

These so-called *few-of-a-kind* assembly production scenarios [7] are typical of SMEs. Since SMEs are the "backbone of the manufacturing industry", e. g. providing some ∼45% of the value added by manufacturing in the European Union [8],

Figure 1: The proposed reconfigurable workcell prototype shown at Hanover fair 2017.

it would be highly beneficial if robotic workcells could be developed specifically to address such use-cases.

The main barrier to greater adoption of robot production in SMEs are the expertise needed in setting up existing solutions and the time for testing and fine-tuning. Since SMEs usually do not have such expertise available, they avoid introducing such solutions, even when they are economically justifiable. We can recognize that these problems are due to the time costs involved in re-configuring and re-programming the robot workcell for new assembly tasks, which are often too prohibitive to make the application of robots profitable.

In this paper, we present the design of a new kind of autonomous robot workcell that is attractive not only for large production lines, but also for few-of-a-kind production. We propose reducing set-up times by exploiting a number of hardware and software technologies, some of which were partially developed in prior work, and some of which are novel contributions in this paper particular to the proposed workcell design.

The main novelty of the workcell lies in the active reconfiguration of passive fixtures and other passive elements in the cell, which can be performed by the robots installed therein. This reconfiguration process allows the robots to autonomously configure their workspace and prepare the workcell for the execution of new assembly tasks.

1.1 Related Work

Many surveys in recent years have followed the development of reconfigurable robotic systems, both in research and in industry [4]. Work of Chen [2] focuses on the modular reconfigurability aspect in particular by finding optimal module assembly configurations from a given set of module components for a specific task. His subsequent work on the design of a reconfigurable robotic workcell for rapid response manufacturing [3] is of particular relevance with respect to the workcell proposed in this paper. However, his work involved a workcell containing hardware elements that can only be reconfigured *manually*. Our proposed workcell focuses on introducing hardware elements that can be actively reconfigured *automatically* by the system itself [5].

In the work of Krüger *et al.* [7,9], a set of methods was developed to facilitate the set-up of complex automated assembly processes. The proposed set of methods included pose estimation and tracking of parts using a 3-D vision system, fast and robust robot trajectory adaptation using *dynamic movement primitives (DMP)* [10], and ROS-based software control and state machine programming. In this work we build on these approaches and add the ability to automatically reconfigure the workcell, while adding a user interfaces to facilitate task set up. The proposed system advances beyond synthetic benchmarks and demonstrates the viability of the system on actual industrial use-case.

2. RECONFIGURABLE SOFTWARE

The introduction of a robotic cell into a production line represents a big investment for SMEs. The high costs usually come from the price of the necessary hardware and the time spent for the integration of the robotic system into the production line. One of the time-consuming aspects is the programming of task sequences for the robots involved in the production process. The programming is normally done via on-line programming using a robot teach pendant connected to the robot controller, or via off-line programming in a simulation environment. Because specific robot system knowledge is required for both approaches, we developed a software system that would facilitate the programming of robot tasks regardless of the robot system. The software system is designed to be distributed, modular and offers seamless reconfiguration of the robot cell. The package also provides the necessary tools to enable simple, intuitive programming of robot tasks.

Our system was build using *Robot Operating System (ROS)* framework, where the hard real-time components were developed using a *Matlab Simulink Real-Time (SLRT)* server, since ROS in its current form does not provide any form of hard real-time implementation. This is a crucial requirement for reliable and accurate robot control.

2.1 System Architecture Overview

Elements of the software architecture of a typical workcell design can be seen in Figure 2. The “Robot Module”s, represent a robot with its robot control SLRT server, one additional measurement unit, and all of its tools and grippers. The “ROS Master Computer” refers to the computer in the system that runs the ROS core and our basic nodes. In order to access various periphery from the workcell a “Digital Interface Unit” is used for bridging the connection from PLC to ROS. Other typical modules include a “Vision Module” and a “Simulation Computer”. The Vision Module represents a user programmable processing unit for typical vision tasks like quality control and part localization. The Simulation Computer offers a dynamic simulation environment provided by the VEROSIM software package, where the production can be planned and evaluated. Depending on requirements, the workcell design can be adapted by adding or removing various modules.

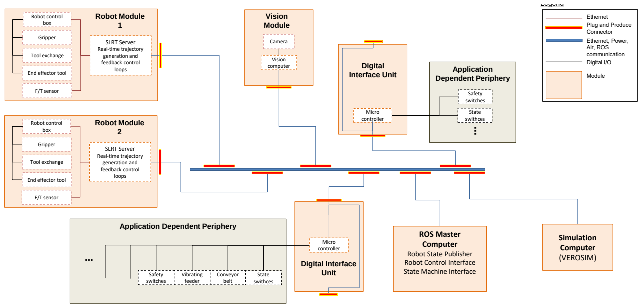


Figure 2: Schematics of the workcell software and hardware architecture.

2.2 Simulink Real-Time Server

The SLRT server, responsible for robot control, connects directly to the robot controller via Ethernet. Standard robot controllers usually only offer basic control methods and not state-of-the-art trajectory generation methods implemented on the SLRT server. Importantly this approach also makes our system independent of the robot. Compatible robots must enable receiving a joint stream over Ethernet and all it takes is a modification of the kinematics model on the SLRT server in order to integrate a new robot. The SLRT server also connects to some other measurement units (e.g. force/torque sensors) that can be used for closed loop control policies (e.g. force control).

The trajectory generation algorithms that have been implemented so far cover the most common robot motion needs in the context of automated assembly. These are: trapezoidal speed profile in joint space; minimum jerk for position and minimum jerk SLERP for orientation trajectories in Cartesian space; admittance force control [1]; joint space dynamic movement primitives for free-form movements [6]; Cartesian space dynamic movement primitives free-form movements in Cartesian space [10].

2.3 ROS Software Package

To allow the robot workcell to be accessed, controlled and calibrated within the ROS environment, various ROS nodes have been developed to offer an interface to the SLRT server and other modules in the workcell.

SLRT State Publisher to read joint positions, velocities, payload, tool information, force/torque sensor data from the SLRT server and publish it via ROS topics using standard ROS messages.

Action Servers handle communication to the SLRT server and are used to trigger robot motion and monitor the progress of the trajectory using *actionlib*. Each trajectory generation algorithm that is implemented on the SLRT server is offered as a separate action server with its own goal, feedback and result messages. The low level logic of all of the action servers ensures that only one motion can be executed at a time.

ROS Services for handling short duration tasks such as changing the state of digital outputs. Our ROS package includes services for changing the robot mode from position control to gravity compensation mode, triggering direct joint control on the SLRT server and setting digital outputs on the robot controller.

Database for keeping track of the workcell state at all times, be it during operation or downtime. We followed a common approach with wide support in the community and implemented a MongoDB database for persistent storage.

Robot Capture Program for capturing and storing various robot related configurations in the database. It is commonly used in conjunction with the kinesthetic guiding of the robot, where the programmer of the robot workcell can freely move the robot in its workspace and then save the points of interest. The functionality is commonly used for calibrating the workcell state (reconfigurable fixture positions, tool pick-up slots) and for saving pick-and-place poses of the robot assembly task.

Programming by Demonstration as an intuitive method for teaching the robot how to move to either points in Cartesian or joint space or over whole trajectories. This method has increased in popularity in recent years and more robot manufacturers are starting to implement the functionality on their robots.

Adaptation of Learned Trajectories by using admittance control. The displacement due to the force error is used as a correcting offset to our DMP encoded trajectory, when the learned trajectory is not ideal or optimal [1].

3. RECONFIGURABLE HARDWARE

The proposed robotic workcell is in large part constructed of modular hardware that allows for fast and easy reconfiguration; from the structural frame to the fixtures, end-effectors, tool exchange system, P&P connectors, and other peripheral devices. With this approach we make it possible to use the proposed workcell in a wide range of industrial applications and environments. Furthermore, we also make it relatively easy to alter its shape and purpose within those environments. The workcell follows the notion that automatic reconfiguration is achieved using robots, which actively reconfigure the passive elements inside the workcell. This concept drives down the cost of the cell making it more affordable for SMEs. The following technologies and solutions were used to achieve said hardware reconfigurability.

Reconfigurable Frame made of rectangular steel beams that are connected via the *BoxJoint* patented modular frame coupling technology. The advantage of this technology is that a workcell frame can be easily configured into a large variety of shapes.

Tool Exchange System for the robots to equip themselves with different grippers needed for different assembly tasks. Tools are introduced into the workcell on trolleys that connect to the P&P connector. If reconfiguration of the cell is needed to assemble a different workpiece, a new trolley with different end-effectors can be introduced into the workcell.

Passive Sensor-less Reconfigurable Fixtures designed in a Stewart platform-esque configuration with six legs, named “hexapod”. These fixtures can be actively reconfigured by the robot arms on demand by connecting a robot to a fixture via the tool exchange system, releasing the fixture brakes, manipulating the fixture into the desired pose, re-applying the brakes, and disconnecting the robot from the fixture.

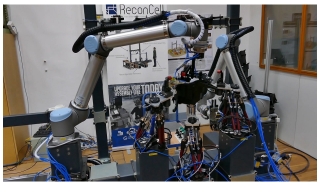
Passive Sensor-less Linear Unit, on which the robot is mounted, to enlarge the work area of the robot within the work cell. The robot is used to propel itself along the linear axis by connecting the end-effector to the frame and moving the base to a new position. Compared to conventional actuated solutions that are much more expensive, this approach is appropriate for applications where the need to move the robot is relatively infrequent.

4. USE CASE EVALUATION

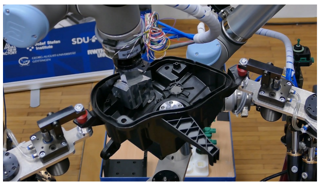
In order to evaluate our proposed robotic cell, we implemented a real industrial use-case. The industrial partner is involved in automotive light production, where the demand for different lights can vary substantially in a single year. The total production of each light housing is typically between 100,000-300,000 units per item. However, these lights are not assembled in one batch. Following the just-in-time production paradigm, a switch from production of one automotive light type to another is often necessary. The company uses product specific assembly devices, which are stored in a warehouse, when not in use. The devices must be stored to produce spare parts for at least the next five years. Production of spare parts in particular is a low quantity production scenario and usually occurs only a few times per year. Each changeover of production lasts several hours and presents a significant cost. It would therefore be extremely useful for suppliers to have a single robot cell available which is capable of assembling many different types of lights, while also being rapidly reconfigurable for alternating production scenarios.

The assembly device are currently operated by people, who insert parts in the machine and check the quality after assembly. Manual work and quality is highly dependent on workers’ qualifications, skills and their knowledge of the assembly process. Customers expect that the supplier company is very flexible in coping with changes in demand. This is why SMEs seek to time every task carefully and look for optimizations. A fully automated assembly procedure also implements quality control checks and integrates in the company’s business intelligence infrastructure providing key performance indicators (KPIs). Defective products can be detected before delivery and the KPIs can be used for analysis and prevention of defects and production optimization.

(a) Active reconfiguration of passive fixtures.



(b) Housing insertion and part pickup.



(c) Part assembly.

(d) Finished product removal

Figure 3: Key production steps for assembly of an automotive light using active reconfiguration.

The technologies described in this paper have been implemented for light assembly in the following way (*c.f.* Fig. 3). Before the start of the production of a new light housing model, the reconfigurable fixtures are actively placed in the appropriate configuration by robots, to accommodate the housing (main body of the headlamp) that comes directly from an injection molding machine. This step happens only once per production of a single type of headlamp (*c.f.* Fig. 3a). In the next step, the robots equip themselves with grippers with which it will pick up assembly parts. The assembly parts are detected using a calibrated visual localization system. The robot places the light housing into the fixture and inserts the remaining parts into the housing one by one (*c.f.* Fig. 3b,3c). After the assembly the robot grasps a camera to inspect the quality of the assembly. Finally, the other robot removes the housing from the fixtures, moving it on to the next step of the production process.

5. CONCLUSION AND FUTURE WORK

In this paper we presented a reconfigurable robotic workcell that targets the manufacturing industry with small production batches where changes in demand happen rapidly. The developed workcell consist of both modular software components and reconfigurable hardware elements. Affordable passive elements are actively reconfigured via robot manipulation to accommodate a different manufacturing process. To demonstrate the benefit of using such a workcell in an actual industrial scenario, a case study has been implemented with a partner from the automotive industry. In our experiments we demonstrated that the developed reconfigurable robot workcell provides the much needed flexibility and fast changeover characteristics for automated assembly processes in the context of automotive lights product family. In the future, we will focus on methods for finding a workcell configuration of reconfigurable components for assembly of a new product automatically, taking into account the constraints of the assembly procedure and the current workcell components.

6. ACKNOWLEDGMENTS

This work has been funded by the Horizon 2020 ICT-FoF Innovation Action no 680431, ReconCell (A Reconfigurable robot workCell for fast set-up of automated assembly processes in SMEs) and by the GOSTOP programme, contract no C3330-16-529000, co-financed by Slovenia and the EU under the ERDF.

7. ADDITIONAL AUTHORS

Additional authors: Žiga Gosar (Elvez d.o.o., Višnja Gora, Slovenia, email: `ziga.gosar@elvez.si`).

8. REFERENCES

[1] F. J. Abu-Dakka, B. Nemec, J. A. Jørgensen, T. R. Savarimuthu, N. Krüger, and A. Ude. Adaptation of manipulation skills in physical contact with the environment to reference force profiles. *Autonomous Robots*, 39(2):199–217, 2015.

[2] I.-M. Chen. *Theory and applications of modular reconfigurable robotic systems*. PhD thesis, California Institute of Technology, 1994.

[3] I.-M. Chen. A Rapidly Reconfigurable Robotics Workcell and Its Applications for Tissue Engineering. `https://dspace.mit.edu/handle/1721.1/3753`, 2003.

[4] M. Fulea, S. Popescu, E. Brad, B. Mocan, and M. Murar. A literature survey on reconfigurable industrial robotic work cells. *Applied Mechanics and Materials*, 762:233, 2015.

[5] T. Gašpar, B. Ridge, R. Bevec, M. Bem, I. Kovač, A. Ude, and Ž. Gosar. Rapid hardware and software reconfiguration in a robotic workcell. In *IEEE The 18th International Conference on Advanced Robotics (ICAR)*, pages 229–236, Hong Kong, China, 2017.

[6] A. J. Ijspeert, J. Nakanishi, and S. Schaal. Learning attractor landscapes for learning motor primitives. *Advances in Neural Information Processing Systems*, pages 1523–1530, 2002.

[7] N. Krüger, A. Ude, H. G. Petersen, B. Nemec, L.-P. Ellekilde, T. R. Savarimuthu, J. A. Rytz, K. Fischer, A. G. Buch, D. Kraft, W. Mustafa, E. E. Aksoy, J. Papon, A. Kramberger, and F. Wörgötter. Technologies for the Fast Set-Up of Automated Assembly Processes. *KI - Künstliche Intelligenz*, 28(4):305–313, Nov. 2014.

[8] European Factories of the Future Research Association. Factories of the Future: Multi-Annual Roadmap for the Contractual PPP under Horizon 2020. Technical report, Publications office of the European Union: Brussels, Belgium, 2013.

[9] T. R. Savarimuthu, A. G. Buch, C. Schlette, N. Wantia, J. Rossmann, D. Martínez, G. Alenyá, C. Torras, A. Ude, B. Nemec, A. Kramberger, F. Wörgötter, E. E. Aksoy, J. Papon, S. Haller, J. Piater, and N. Krüger. Teaching a Robot the Semantics of Assembly Tasks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2017. doi: 10.1109/TSMC.2016.2635479.

[10] A. Ude, B. Nemec, T. Petrič, and J. Morimoto. Orientation in cartesian space dynamic movement primitives. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 2997–3004, Hong Kong, 2014.