Title:	A Reconfigurable robot workCell for fast set-up of automated assembly processes in SMEs
Acronym:	**ReconCell**
Type of Action:	Innovation Action
Contract Number:	680431
Starting Date:	1-11-2015
Ending Date:	31-10-2018

Deliverable Number:	D1.2
Deliverable Title:	Demonstration of optimal workcell reconfiguration in simulation
Type (Public, Restricted, Confidential):	PU
Authors :	M. Deniša, M. Priggemeyer, E. Guiffo Kaigom, D. Kaufmann, A. Ude, and J. Roßmann
Contributing Partners:	MMI, JSI

Estimated Date of Delivery to the EC:	31-10-2017
Actual Date of Delivery to the EC:	15-11-2017

Contents

1 Executive Summary

This deliverable presents two demonstrations to show optimal workcell reconfigurations in simulation. The first demonstration deals with robot cell construction, path generation and passive fixture reconfiguration. The second demonstration presents position optimization of various cell elements w.r.t. robot's energy efficiency. Two videos presenting the respective cases are attached to the deliverable. Both of them are further described in the following two sections.

2 Workcell reconfiguration in simulation

The first video attached to this deliverable is available at

`https://abr-svn.ijs.si/ReconCell/Deliverables/D1.2/D1.2_Demo1.mp4`

It demonstrates the creation of workcell designs and reconfiguration of existing layouts in the simulated environment. The attached video example presents the powerful tools of the simulation system, which enable a user-friendly and efficient workcell reconfiguration. Core elements include the creation and manipulation of poses and paths in relation to the Visual Programming with ActionBlocks [4] and inclusion of general cell equipment and specific use case objects from the ReconCell ModelLibrary. In general, the video presents the creation of a minimal cell design with an actuator and a passive fixture (i.e. Hexapod). After demonstrating the placement of beams and other parts, a reconfiguration is visually programmed by the operator.

2.1 Cell configuration

Initial cell designs can be configured easily by applying different tools in the simulation software. The integrated Model Library was extended to provide ReconCell specific models. Individual model properties are further described in deliverable D4.3. The Model Library is divided into categories in order to ease the navigation and find the location of needed cell elements. Core cell equipment includes parts like beams in different dimension and sizes and the corresponding box joints to connect them. In addition, non-static cell elements like the passive fixtures (i.e. Hexapods) and actuators (UR10 robots) are provided, that can either be actively controlled or passively guided to user-definable poses. Finally, elements specifically designed or needed in the ReconCell use cases are also provided in the ModelLibrary. These include feeders to present parts to an actuator, grippers to pick up one or more parts, and fixtures to support work pieces during an actual assembly process.

Since all parts in the Model Library are equipped with Docking Ports, each one can be easily handled in the 3D simulated environment. Besides the option to rotate and position parts in a constrained fashion, two parts can be snapped together by selecting their respective Docking Ports. Constraints help to place parts accurately at desired poses by limiting the degree of freedom for movement operations (e.g. allow in-plane movements or rotation around specific axes).

2.2 Application for cell reconfiguration

At some point during the design process of an assembly sequence, the cell or parts of it might be subject to change. This might happen due to changes in work pieces or parts better applicable to the process or because of optimizations in spatial work cell usage (i.e. limitations) or in the actuation of parts (i.e. energy efficiency, reachability). These static reconfigurations can be accomplished by the same means as they were used during the initial cell design.

Another reason to reconfigure the cell is to make a changeover from one assembly process to another without changing beams or base positions of parts [3]. This reconfiguration will only include passive fixtures that can be automatically reconfigured by utilization of an actuator. Hexapods provide a flexible way of providing fixtures for work pieces [2]. The hydraulic lock and release mechanism can be triggered to move the hexapod top plate to a new position before fixing the mechanics in that new position.

The same feature is available in the simulated environment and demonstrated in the attached video. After the placement of parts in the cell is done, poses are taught by guiding the actuator to different positions. This pose list is automatically converted to an ActionBlock network that is afterwards manually extended to lock and release the tool exchange system. Figure 1 depicts the execution of a simulation run reconfiguring the Hexapod.

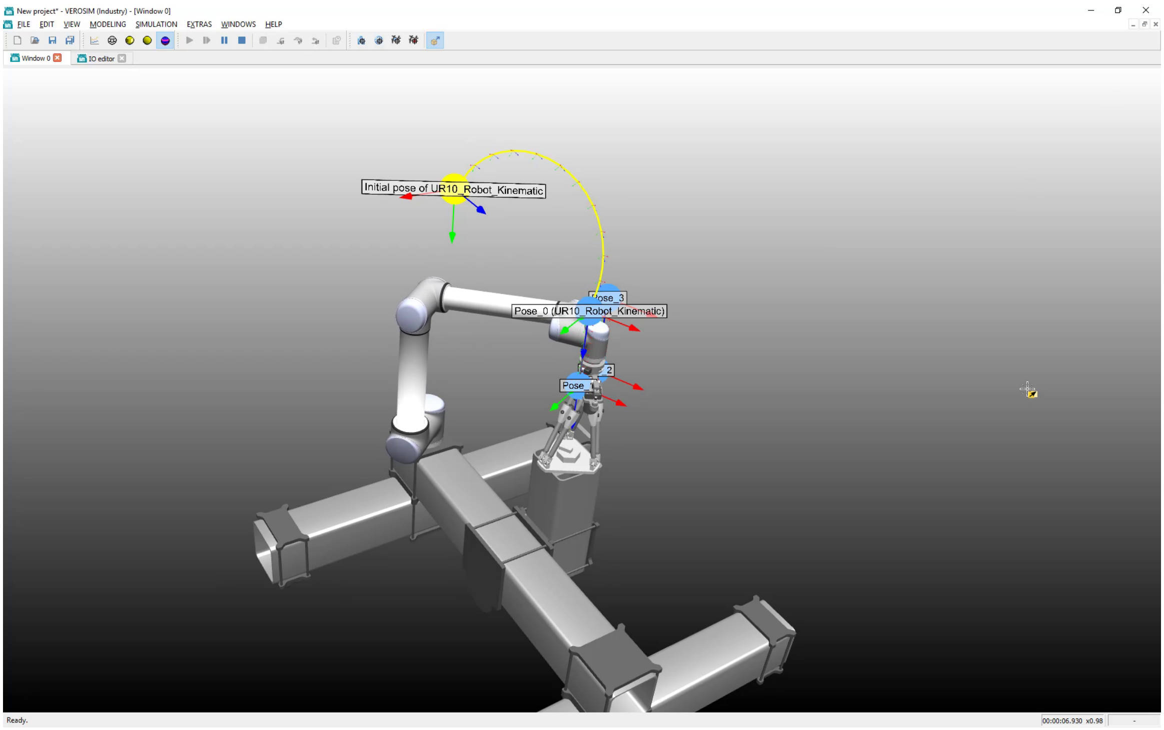

Figure 1: Simulated reconfiguration of Hexapod top-plate poses.

3 Energy efficiency optimization

The second attachment is a video demonstration showing basics for workcell optimization in a simulation environment. It is available at

`https://abr-svn.ijs.si/ReconCell/Deliverables/D1.2/D1.2_Demo2.mp4`

One of the major optimization requirements, beside cycle time, is often energy efficiency of the cell [1]. In the video, we show two examples of position optimization w.r.t. robot's energy efficiency.

While the first part shows trolley displacement optimization in one dimension, the second one deals with a robot base position optimization in two dimensions. Parameter spaces are defined and varied by simulating different possible cell configurations and states. They can be defined as discrete and quasi-continous parameters. While the first example uses quasi-continous parameters, i. e. trolley displacements, the second optimization example uses discrete parameters, i. e. robot base positions. Parameter sets are easily selected by the user from a preprepared list of sets. In addition, different points in the simulation time can also be selected. Alternatively, the user can select a parameter set at a specific point in time.

The estimation of the robot's current energetic state is based on the torque generated by the drivetrain inside each joint. Based on the relation between motor torque and joint velocity, the actual energy level for each joint is calculated. The difference between consumed and regenerated energy is determined based on the evaluated powerflow direction. The accumulated consumed energy of the robot forms a metric for further optimization of object placement and cell reconfiguration. As seen in the video, the simulation environment enables the visualization of the outcome w.r.t. the simulation stage. For example, current accumulated energy

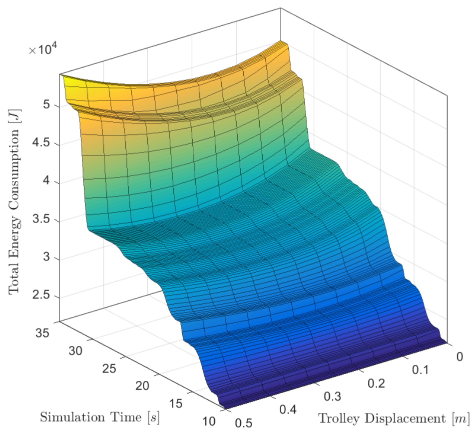

Figure 2: Example data gained in simulation for trolley displacement optimization. Plot shows robot's total energy consumption over the simulation time w.r.t. trolley displacement. We can observe a local minimum which can be applied to the cell.

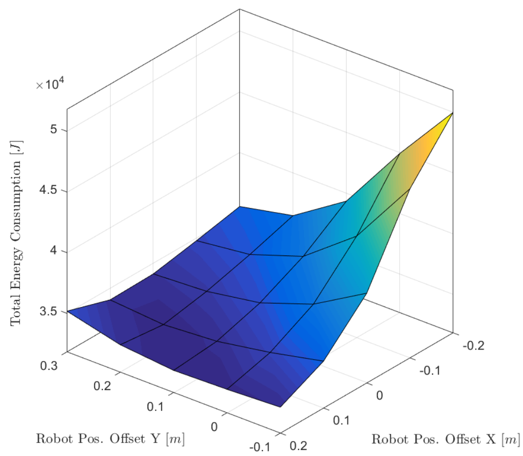

Figure 3: Example data gained in simulation for robot base positions optimization. Plot shows robot's total energy consumption for task execution w.r.t. different robot base positions.

consumption can be represented as a robot color.

Results of the simulation can be exported and used for further analysis. In addition to the video, results for both examples are also presented in Fig. 2 and 3.

References

[1] L. Atorf, C Schorn, C. Schlette, and J. Roßmann. "A Framework for Simulation-based Optimization Demonstrated on Reconfigurable Robot Workcells". In: *IEEE International Symposium on Systems Engineering (ISSE)*. Vienna, Austria, 2017.

[2] M. Bem, M. Deniša, T. Gašpar, J. Jereb, R. Bevec, I. Kovač, and A. Ude. "Reconfigurable fixture evaluation for use in automotive light assembly". In: *18th International Conference on Advanced Robotics (ICAR)*. Hong Kong, 2017, pp. 61–67.

[3] 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: *18th International Conference on Advanced Robotics (ICAR)*. Hong Kong, 2017, pp. 229–236.

[4] D. Losch and J. Roßmann. "Visual Programming and Development of Manufacturing Processes Based on Hierarchical Petri Nets". In: *3rd International Conference on Soft Computing & Machine Intelligence (ISCMI)*. 2016, pp. 154–158.