

Open Access

Cooperating Robot Force Control for Positioning and Untwisting of Thin Walled Components

Bertelsmeier F*1, Detert T2, Übelhör T1, Schmitt R1 and Corves B2

¹Chair for Metrology and Quality Management, RWTH Aachen University, Steinbachstr, Aachen, Germany ²Department of Mechanism Design, Dynamics of Machines and Robotics, RWTH Aachen University, Kackertstr, Aachen, Germany

Abstract

Geometric deformations in large thin walled components such as airplane shells caused by gravitational influences have to be compensated before assembly. The research objective is to develop a metrology assisted robot based assembly system that detects deformations and compensates them through force controlled movements of the robots. This paper describes the development of a cooperating robot force control for three industrial robots. The control algorithm uses a numerical optimization to take into account the strong mechanical coupling of the robots by the handled component.

Keywords: Robotic; Metrology; Automation; Assembly; Selfoptimization

Introduction

In order to position and shape large components in assembly, flexible programmable devices like industrial robots have to be capable of fitting components into their desired shape considering position tolerances. Particularly in the case of non-rigid lightweight structures such as airplane shells [1] the compliance with the nominal geometric shape and tolerances cannot be achieved with cooperating robots using standard control approaches.

The reason for this is that these components can deviate significantly from their nominal geometry. The deformations result from the weight of the components, temperature fluctuations and manufacturing tolerances. These influencing variables lead to individual deviations for each component - each component is "unique" [2]. The handling of these elements needs advanced control and modeling approaches, based on a permanent metrological feedback for the complete handling system.

In order to solve these challenges and to create a flexible robot based handling system for large components, the research project "Self-optimizing Assembly Systems" as part of the Cluster of Excellence at RWTH Aachen University ("Integrative Production Technology for High Wage Countries") is developing an assembly system with cognitive abilities to achieve the necessary geometry and positioning tolerances. The aim of the subproject is to enable assembly systems to react independently to process deviations and environmental conditions and to initiate compensation strategies. In case of the handling of large, components the assembly systems must be able to identify individual geometry deviations and to autonomously create control strategies for compensation.

This paper focuses on the development of a cooperative robot force control for the handling of large thin-walled components based on a numerical optimization enhanced control algorithm. At the beginning of the paper the state of the art concerning the handling of components with cooperating robots is summarized. Under consideration of the deficit of the existing solutions, the use case and the developed handling system for the depicted project are explained. For the development of the control strategy, the limitations and requirements as well as the system behavior of the control system are defined. Based on these boundary conditions the new control strategy for cooperative robot force control is developed and in the final step validated using the demonstration.

State of the Art

In the following chapter the existing methods for force control of cooperating robots are presented. The existing developments can be split into two main approaches, the cooperative force control for rigid objects and the development of cooperative force control for non-rigid components.

Cooperative force control of rigid objects

Hayati [3] uses a hybrid force position control, which is based on Craig and Raibert [4] and uses the assumption of a rigid body. For any point on the surface of the component, a hybrid force and position state can be defined. The method is constructed with two separated control systems, one for the force control and the other one for the position control. The disadvantage of Hayati's approach is that the dynamics of the entire system must be known. This includes both the inertia of the robots and the inertia of the object to be handled [5].

Another method for force control of rigid components is given by Alberts and Soloway; they propose an extension of the 'kinematic resolved-rate control'. With this method only the kinematics of the robot must be known [5].

Cooperative force control for non-rigid components

Flexible objects lead to the problem that no defined closed kinematic chain can be formed. In addition, flexible objects are generally vibrating and thus cause additional forces in the system, which must be

*Corresponding author: Felix Bertelsmeier, Chair for Metrology and Quality Management, RWTH Aachen University, Steinbachst, Aachen, Germany, Tel: +49-241-80-20584; E-mail: F.Bertelsmeier@wzl.rwth-aachen.de

Received November 15, 2017; Accepted November 23, 2017; Published November 27, 2017

Citation: Bertelsmeier F, Detert T, Übelhör T, Schmitt R, Corves B (2017) Cooperating Robot Force Control for Positioning and Untwisting of Thin Walled Components. Adv Robot Autom 6: 179. doi: 10.4172/2168-9695.1000179

Copyright: © 2017 Bertelsmeier F, et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Citation: Bertelsmeier F, Detert T, Übelhör T, Schmitt R, Corves B (2017) Cooperating Robot Force Control for Positioning and Untwisting of Thin Walled Components. Adv Robot Autom 6: 179. doi: 10.4172/2168-9695.1000179

Page 2 of 7

compensated. The handling task of non-rigid components can be split in two main methods [6].

The first method is the targeted deformation of a flexible object by two robots. One approach for this method is the use of the finite element method (FEM) as proposed by Kosuge et al. [7]. The method does not include an exact analytical model to describe the object, since such a model is rarely available. The FEM-Model is used to calculate the necessary internal moments for a given deformation of the considered object [8].

The other method is the positioning of a flexible body. The work of Sun et al. [9] is designed to divide the dynamics of a flexible body into a rigid and a flexible component. The dynamics are described as "Clamped-Free" [8]. This means that the object is rigidly clamped on one side; the other side is free and is loaded with external forces. In the implementation for robots, this means that the robot of the rigid clamping is purely position-controlled.

Deficit of the state of the art

Most of the methods described aim at distributing the load as widely as possible to all robots when handling components with several robots. Only one approach describes a method that allows the targeted deformation of a component via cooperating robots. The disadvantage of this method is the use of a FEM-Model for the calculation of the robot forces, which leads to long calculation times for complex components and therefore cannot be used during production processes.

Handling System for Airplane Shell Elements

In this research project the handling of airplane shell elements is used as a demonstration process for the integration of a flexible robot based handling system. The task of the handling systems is to hold the component according to its geometrical shape, required for further assembly processes.

The system has to perform the two steps positioning and untwisting of the component. For the positioning, the component is moved into the desired position and orientation without relative movement between the kinematics. For this movement, all kinematics must be controlled cooperatively to achieve a constant movement of the shell without any deformations [9]. The second step is to untwist the shell to achieve the required shape of the component by individual control of the single kinematics. The compensation movements are calculated according to the measured geometrical deviations.

The experimental setup for the positioning and untwisting process of the airplane shell consists of three industrial robots equipped with force/torque sensors and a shell supported by an aluminum frame (Figure 1). The demonstrator is equipped with a Laser tracker measurement system and force/torque sensors on each robot. In order to not limit the possible deformations of the shell, the shell is pivotally suspended at its upper edge and loosely at the bottom. The test component is a downsized CFRP shell with stringers integrated (1.7 m \times 2.1 m). The robots and the applied force and torque values are referred as robot/force/torque 1 to 3, starting with the bottom robot.

In order to be as flexible as possible to apply forces at different points on the shell, three robots from Mitsubishi are used, which are equipped with external force torque sensors. The force sensor system enables the measurement and regulation of the acting forces. For each robot, Mitsubishi offers an internal force control that implements a hybrid force, stiffness and position control. The force sensors are designed for a nominal load of 200 N and 4 Nm. Statically up to 1000



N and 6 Nm can be measured. The robot controllers belong to the Mitsubishi Q series, which allows the use of a PLC, which facilitates the cooperative operation by collision avoidance and a common variable storage. The robots are arranged vertically on a tower to deform the aircraft shell in the plane normal to the stringers. Pneumatic grippers are used to hold the shell. Due to the fact that the robots work together, a common world-coordinate system must be introduced and placed in the base of the robot 1. The coordinate transformations between the robots coordinate systems have been initially determined using a Laser tracker.

For the continuous determination of the component geometry and the deformation state, the same Laser tracker is used. It measures several predefined points on the shell in order to obtain discrete 3d-point coordinates on the panel for the determination of the individual component deformations. For the calculation of the geometric deformations the single measurement points are compared to the nominal geometry of the shell (CAD-Model). Based on this information the system is capable of calculating the necessary compensation forces that need to be applied by the robotic system [10]. The system uses a model-based process control that utilizes a model of the component deformation behavior to analyze the component individual deformations and calculate force and torque values for compensation. The method and concept for the model-based process control is described by Borrmann et al. [10].

Requirements and Limitations of the Control System

The challenge of the force control is the strong mechanical coupling of the robots, so the robots cannot be considered separately. In addition to the coupling, it must be considered that the final state of the control must be static. The analysis and description of the system is the first step in order to derive the control strategy.

The shell is pivotally suspended at its upper edge in the longitudinal direction (Figure 2). The forces within the joints cannot be measured directly and should not contribute to the deformation of the shell. The shell itself is stiffened by stringers in the longitudinal direction and it can be assumed that no deformation takes place in the direction of the stringers. Consequently, the handling process can be considered as a planar problem.

The aircraft shell is deformed by applying defined forces. If however, only forces are given, they can in principle be arbitrarily oriented in space. In addition, acceleration of the entire system occurs even with small deviations from the equilibrium of forces or torques. With a known weight and center of gravity of the component, this can





be compensated by a corresponding force specification. Due to the fact that the center of gravity of the aircraft shell is unknown, torques and forces caused by the weight distribution cannot be compensated continuously by the controller. Therefore, it is also necessary to limit the problem geometrically. The easiest way is to use one robot as a fixed clamp. This means that one robot is purely position-controlled and absorbs all deviating forces. The forces acting on this robot are indirectly given by the other two robots. By using this method, the control strategy can be easily extended to a system that does not use fix jigs.

The robot configuration offers an internal force control for each robot. Each of the three robots is capable of applying defined forces, by utilizing the attached sensor. This control routine is not capable of handling the mechanical coupling between the three robots, so an overall control routine is still necessary for the complete system; however the internal force control can be used as a subordinated control loop to move the robot (Figure 3). The alternative is an external force control, in which only the positions of the robots can be specified. Consequently, the force could only be adjusted by using the virtual stiffness coefficients. This would lead to an even more complex system and complicate the task unnecessarily.

For the internal force control the robots have a maximum permissible payload of 7 kg and maximum torque of 6.86 Nm. However, static operation loads can exceed this value significantly. The maximum load of the sensors is 200 N and the acting torque cannot exceed 4 Nm. Therefore, the maximum permissible load of the sensor is used as a limitation for the maximum applied load by one robot. This limitation is directly configured in the robot controller.

Based on the limitations and system features, some assumptions must be defined to enable the development of the cooperated robot force control. These assumptions are:

1. No deformation in the stringer direction: The stringers in the longitudinal direction are much stiffer than the thin skin. Therefore, it is assumed that the deformation will be mainly perpendicular to the

stringers. The problem can be reduced to a flat problem (Two degrees of freedom).

Page 3 of 7

2. Quasi-static: To avoid damages to the shell from dynamic loads, the forces and torques are applied slowly. Consequently, dynamic loads of the system are not considered in this work.

3. Linearity for small changes: In each case, small geometry changes are carried out during the control, these can generally be approximated by a straight line in the corresponding operating point. The system is thus approximated by coupled linear springs.

4. Sufficiently exact internal force control of the robots: If the internal force control of Mitsubishi is used, it is assumed that this can control the applied force with sufficient precision within the scope of sensor accuracy.

Due to the previously mentioned assumptions, the system can be simplified as a planar model of coupled springs (Figure 4).

The upper robot serves as a fixed joint and can only be influenced indirectly by the other two robots. The other two robots represent free forces. For the static system, the speed and acceleration terms are zero. This allows describing the system by means of a static equilibrium. With robot 3 as a fixed joint, equation (1) describes the static system.

$$\begin{pmatrix} F_{3} \\ \hat{o}_{3} \end{pmatrix} = - \begin{pmatrix} F_{1} + F_{2} \\ \hat{o}_{1} + \hat{o}_{2} + r_{31}(t) \times F_{1} + r_{32}(t) \times F_{2} \end{pmatrix}$$
(1)

The equation of the static equilibrium is used during the control routine as base equation for the optimization problem.

Development of the Control System

Most design methods for a control strategy require the previous modeling of the control system. In case of the airplane shell deformation such an analysis is only possible for some parts of the system since no exact model of the shell exists yet. The long-term goal is to train such a model with measured data obtained by using the cooperative force control [10]. Until then, no methods for the design of the control system can be used that require an exact description of the control path.

Another basic problem is that the control system should produce a static state of the shell to use the equation outlined above. For this purpose, it is necessary to apply geometric limitations, which uniquely define the position of the shell in space. The forces could otherwise lead to a dynamic system, if small deviations from the force equilibrium occur. Consequently, one robot must hold its position and cannot



react directly to changes of the external force. The acting force son this robot can only be influenced indirectly by the forces of the other two robots. Under these conditions the system is undefined, since only two robots have to provide a force state for three points. One solution is to consider only two robots and ignore the third robot. The forces of the two moving robots must be set precisely so that the force of the third robot results from these two forces. In this procedure, the force acting on the third robot cannot be considered. Interference variables such as friction in the bearing points can therefore not be compensated and the calculation model for setting the forces is not accurate enough. Instead, the goal is to develop a method which can continuously use the measured values of all three robots in order to reach the required state of force.

Parallel Force Value Setting with Optimizer

With a parallel force value setting, the measurement data of all robots are continuously taken into account. However, when the internal robot-force control is used in parallel, the system might develop dynamic system behavior in which the robots oscillate. To avoid oscillations in the system, appropriate admittance and damping factors must be selected when using the internal force control system.

Since the upper robot remains position-controlled, the force of this robot must be defined by the other two robots. Assuming a quasistatic system, the static equilibrium (see section 3) is reformulated into a matrix notation Z(t). This matrix allows for specifying a force for the third robot without knowing the elastic properties or mass distribution of the shell. The resulting equations are similar to the equations from Williams work on cooperative force control [11]. Williams also assumes the existence of a static equilibrium, but the resulting equations are used to approximate internal forces and not to control a given total force state. The matrix of the quasi-static equation for the two-dimensional case is equation (2):

$$F_{stat} = Z(t)f \tag{2}$$

$$Z(t) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & -1 & 0 \\ 0 & -r_{31,z}(t)r_{31,x}(t) - 1 - r_{32,z}(t)r_{32,x}(t) - 1 \end{pmatrix}$$

F_{stat}=Resulting static force

Z(t)=Matrix of static equilibrium

The distribution of the forces to the robots can be accomplished by solving an optimization problem. The deviation between the possible static force state and the desired force state needs to be minimized. A minimization of the least squares is used to consider normal deviations and to punish individual large deviations. Additional a weighting matrix is introduced to make the torque comparable with forces.

$$\begin{split} & \min_{F_{stat}} \left(F_{stat} - F_{des} \right)^T W \left(F_{stat} - F_{des} \right), \text{ with } w_{ij} > 0 \\ & \Rightarrow \min_{f} \left(Z(t) f(t) - F_{des} \right)^T W \left(Z(t) f(t) - F_{des} \right), \text{ with } w_{ij} > 0 \\ & F_{stat} = \text{Resulting static force} \end{split}$$

F_{des}=Desired force state

W=Weighting matrix

f=Solution vector of the optimization for robot 1 and 2

The solution to this problem is the vector f, which represents the force specification for robot 1 and robot 2. The matrix W can be used to weight the resulting torque values in the optimization in comparison to the larger forces. It is also possible to weight normal and tangential forces differently, but for this a transformation from the tool coordinates into the world coordinates is necessary. The matrix Z changes continuously due to the deformation and must be determined for each step. The ongoing calculation of the matrix Z and the implementation of the optimization can be very computationally intensive. By assuming a constant matrix Z, the computational effort can be reduced using a suitable optimization algorithm.

The optimization equation does not take into account any disturbance variables and is only usable for pure set point specification. In order to be able to compensate disturbance variables, a feedback of the current force state is necessary. Figure 3 shows the resulting control loop for the cooperating robot force control. The control deviation is determined by the subtraction of the actual force state with the desired force state. A standard PI-Controller generates the manipulated variable ΔF from the control deviation. This variable is transferred to an optimizer, which considers the static behavior of the system and thus the indirect presetting of the force for the third robot. Result of the optimization is the manipulated variable Δf for the robots 1 and 2. The robots control the specification Δf by means of the internal force control.

This construct is not a control circuit in the actual sense, since the input and output of the system are the same magnitudes. The system must be strictly defined as set point specification. The actual force control is done within the internal robot-force control.

Implementation of the Parallel Force Value Setting

The force control is performed by the internal Mitsubishi force control of the robots. The robots receive the force specification and the force control mode from a central process control via the Data Link interface. In the robot program itself, the parameters for admittance and damping are already specified to avoid further delays due to the slow network interface. The admittance must be low and the damping must be active, otherwise the robots would oscillate during the parallel force control.

The implemented procedure uses a loop to realize the control algorithm. The approach is based on the use of an system of equations which describes a static force state by an equilibrium of forces and torques (see section 4). The system of equations includes the distance vectors between the robots to take into account the torques created by the leverage between the robots. Due to the deformation of the shell, the distance vectors constantly change, so they are determined in each iteration run by a query of the current position. The linear system of equations can be formulated using the Matlab Symbolic Toolbox and converted into a matrix. The generated matrix can then be transferred to an optimization function. As mentioned in section 4, a weighting matrix is used in order to make the critical torques comparable with the forces. Due to the fact that the forces with a maximum of 200 N can be selected to be significantly greater than the torques with a maximum of 4 Nm, a high weighting of the torques is necessary. However, this high weighting can lead to a misdirection of the optimization problem, since the forces may deviate in a larger region. In particular, for high loads

Page 5 of 7

it is useful to keep the torques of each robot constant to zero so that the limit is not exceeded and the weights can be selected from a wide range. The forces of the robots are still calculated by the use of the static equations keeping the equilibrium state [12].

In the development of this method, a controller is applied, which provides the input for the optimizer. It is implemented as a discrete PI controller so that permanent deviations can be compensated.

As described in section 4, deviations are optimized in each iteration run and added to the actual state after the optimization. The optimizer and the control algorithm are programmed in Matlab. By default Matlab offers the function lscov (A, B, w) to determine a solution in the sense of the least squares with weights. This optimization does not necessarily keep the signs of the actual deviation. The permissible force values may be exceeded and thus damage the component. In addition to the lscov algorithm the Matlab Optimization Toolbox offers the option of optimizing with lsqlin (C, d, A, b). This function solves the problem of equation (3):

$$\min_{x} \frac{1}{2} \left\| C \cdot x - d \right\|_{2}^{2} \text{ with } A \cdot x \le b$$
(3)

A = Matrix of the inequality

b = Vector of inequality

C = Matrix of the linear equation system

d = Set point vector

x = Solution vector in the sense of the least squares

The restriction by the inequality can be used so that the signs which result from $C \cdot x$ correspond to the sign of d. The vector b is chosen as the zero vector. The following relation holds for the inequality.

 $A \cdot x \le 0 \Leftrightarrow -A \cdot x \ge 0$

Now the lines from C to A are transmitted with the following selection of the signs.

$$A_{ii} = C_{ii}$$
, for $d_i \leq 0$

 $A_{ij} = -C_{ij}$, for $d_i > 0$

Consequently, the signs of the optimized solution correspond to the sign of the setpoint. A disadvantage of this method is the poor computing performance. Due to this fact it is not suitable for implementing a real-time control system, especially on weaker hardware in the form of an embedded system.

Apart from the lsqlin method the optimization problem can be solved by using the Moore-Penrose pseudoinverse matrix. This solves undetermined linear equations in the sense of the least squares error. On the assumption that the deformations are only small compared to the size of the shell, it is sufficient to determine this matrix only once before the start of the control process. The optimization problem is thus reduced to a simple matrix-vector product and requires less processing power. Using this method, it still has to be taken into account that force and torque magnitudes can run out of the permitted range. During the project both optimization strategies were implemented and compared to ensure repeatability.

Validation of the Cooperative Robot Force Control

For the validation of the robot force control it has to be considered, that the deformation behavior of the shell is unknown and therefore no analytical calculation of the shell deformations according to the applied forces is possible. This leads to the problem, that the deformation of the panel due to the force values of the robots cannot be compared to nominal values. Another problem at the current state of the development is that the desired force state has to be chosen by the operator. Without the knowledge of the mechanical system it is nearly impossible to define forces that allow a static state of the shell. For this reason, the control algorithm can never reach exactly the required force state and must therefore approach the specified values optimal distributed to the robots. Nevertheless, in order to validate the functionality of the control algorithm, the reproducibility of the two optimization methods and the internal force control was analyzed.

Optimization with Lsqlin

Figure 5 shows the typical trend of the forces for the lower robot



using the parallel setpoint specification based on the Lsqlin optimizer with a position-controlled upper robot. The force setting for this robot was 10 N in the negative X direction.

The control value starts with -20 N in X- and Z-direction. These starting values result due to the weight of the shell. Robot 1 at the bottom must pull the shell away from the frame. Therefore, the robot has to apply -21.5 N in X-direction.

The setpoint specification distributes the disturbance variables on the three robots equally by using the least square method. The control algorithm is very robust due to the return of the current positions and force values and does not tend to overshoot. The lsqlin function has a poor performance, so that transmission of the force control values to the robot controller is slow and not smooth. The reproducibility of the final control values in this method depends on the choice of the parameters for the controller and optimizer. The factor of the P component must be chosen to be less than 0.5 so that the deflection is not too fast. The I portion should not be used, as this leads to very different final states. The weighting leads to poor conditioning of the problem and thus also to high deviations. Therefore, the weights should be selected as evenly as possible.

Optimization with Moore-Penrose pseudo inverse

Figure 6 shows the characteristic trend for the optimization with the Moore-Penrose optimization strategy. The forces and torques are derived from robot 2, which should apply -60 N in X-direction.

The graphs show that the force values have a much smoother run in comparison to the lsqlin optimizer. This is achieved by the more frequent transmission of the force control values to the robot controllers.

Comparison between both methods

Both the use of the lsqlin function and the use of the pseudoinverses have the potential to control the final force state with a high reproducibility. During experiments, the same state with the same parameters was applied four times for each optimization method. For each experiment, the following forces were given in the X-direction: robot 1 (-10 N), robot 2 (-20 N) and robot 3 (10 N). For the lsqlin method, the parameters were chosen to be as robust and reproducible as possible. The second series of experiments using the Lsqlin method uses a high P-component (1) and high weights. The proportionality factor 1 was chosen for the use of the pseudo-inversions. The weightings and integrating behavior were not used in this method (Figure 7).





Figure 7: The results show that both methods lead to similar final deviations. However, for higher forces, it was not possible with the lsqlin method with fixed clamping to carry out the control robustly and to record useful measuring results since the maximum load (200 N) was regularly exceeded. Therefore, the Moore-Penrose pseudoinverse will be used for the flexible handling system.

Summary and Conclusion

The cooperating robot force control for the handling of large thin walled components is developed and put into operation within the scope of this work. The force state of the respective robot is controlled by the internal robot force control, which has been configured in such a way that the force control takes place without system overshoots. The mechanical coupling between the robots is compensated by a new Parallel-Force-Value Setting-Method. This method was validated in particular for the reproducibility of the results. It has been found that a parallel presetting of the forces leads to reproducible results. Using an optimizer, it is possible to reproducibly specify an arbitrary force state for the robots even though the system is undefined and is constantly changing due to deformations. It has been found that the reproducibility can be further improved by utilizing different methods for solving the optimization problem. The actual method used for the optimization is the Moore-Penrose Pseudoinverse which leads to fast and smooth behavior of the control values.

Acknowledgement

Adv Robot Autom, an open access journal

ISSN: 2168-9695

The depicted research has been funded by the German Research Foundation DFG as part of the Cluster of Excellence "Integrative Production Technology for High-Wage Countries".

References

- 1. Engmann K (2008) Technology of the aircraft. Bird book publisher, pp: 336-346.
- Fraunhofer-Institut f
 ür Fertigungstechnik und Angewandte Materialforschung (2012/2013): Automatisierte FVK-Gro
 ßstruktur-Montage – Fraunhofer FFM

entwickelt grundlegende Verfahren und Anlagen. In: Jahresbericht editor. S98-S106.

- Hayati S (1986) Hybrid position/force control of multi-arm cooperating robots. In Robotics and Automation Proceedings 3: 82-89.
- Craig JJ, Raibert MH (1979) A systematic method of hybrid position/force control of a manipulator. In Computer Software and Applications Conference, pp: 446-451.
- Alberts TE, Soloway DI (1988) Force control of a multi-arm robot system. In Robotics and Automation Proceedings IEEE International Conference, pp: 1490-1496.
- Ramirez J, Wollnack J (2014) Flexible automated assembly systems for large CFRP-structures. Procedia Technology 15: 447-455.
- Kosuge K, Yoshida H, Fukuda T, Sakai M, Kanitani K (1995) Manipulation of a flexible object by dual manipulators. In Robotics and Automation Proceedings, IEEE International Conference 1: 318-323.
- Barker LK (1983) Kinematic equations for resolved-rate control of an industrial robot arm. NASA Technical Memorandum.
- Sun D, Mills JK, Liu Y (1998) Hybrid position and force control of two industrial robots manipulating a flexible sheet: Theory and experiment. Robotics and Automation Proceedings International Conference 2: 1835-1840.
- 10. Borrmann C, Brencher T, Ramirez J, Wollnack J (2013) Robot manufactures aircraft parts.
- Schmitt R, Corves B, Loosen P, Brecher C, Jeschke S, et al. (2017) Cognition-Enhanced, Self-optimizing Assembly Systems. Integrative Production Technology (Springer), pp: 877-990.
- Williams D, Khatib O (1993) The virtual linkage: A model for internal forces in multi-grasp manipulation. Robotics and Automation Proceedings International Conference, pp: 1025-1030.

Page 7 of 7