An Adaptive Control Approach to Robotic Assembly with Uncertainties in Vision and Dynamics

Emir Mobedi*, Nicola Villa*, Wansoo Kim, and Arash Ajoudani

Abstract— The objective of this paper is to propose an adaptive impedance control framework to cope with uncertainties in vision and dynamics in robotic assembly tasks. The framework is composed of an adaptive controller, a vision system, and an interaction planner, which are all supervised by a finite state machine. In this framework, the target assembly object's pose is detected through the vision module, which is then used for the planning of the robot trajectories. The adaptive impedance control module copes with the uncertainties of the vision and the interaction planner modules in alignment of the assembly parts (a peg and a hole in this work). Unlike the classical impedance controllers, the online adaptation rule regulates the level of robot compliance in constrained directions, acting on and responding to the external forces. This enables the implementation of a flexible and adaptive Remote Center of Compliance (RCC) system, using active control. We first evaluate the performance of the proposed adaptive controller in comparison to classical impedance control. Next, the overall performance of the integrated system is evaluated in a peg-in-hole setup, with different clearances and orientation mismatches.

I. INTRODUCTION

Robotic applications play a prominent role in industry, where repetitive tasks can be performed faster and cheaper in comparison to manual layouts. Early applications of robots in industry situated them in closed and isolated workcells to avoid physical collisions with human workers and the consequent injures. Nowadays, however, robots can enter in direct contacts with humans, due to their adaptive hardware and software characteristics that ensure compliance with several safety standards such as ISO 10218-1. The use of elastic elements such as springs [1] in the mechanical design of the robots, or the implementation of advanced control strategies (e.g., impedance control) to render variable output dynamics [2] are common examples to achieve adaptive and safe operations for robots in unstructured and hybrid (human populated) environments.

Peg-in-hole like assembly actions is well-known scenarios where the application of robots with the aforementioned characteristics has been studied since 1980s [3], [4]. In such tasks, the uncertainty sources arising from the planning and perception [5] contribute to increasing the complexity of the task execution from a robotics point of view, which is mainly due to the misalignment of the peg with the hole. Passive approaches based on the remote center of compliance
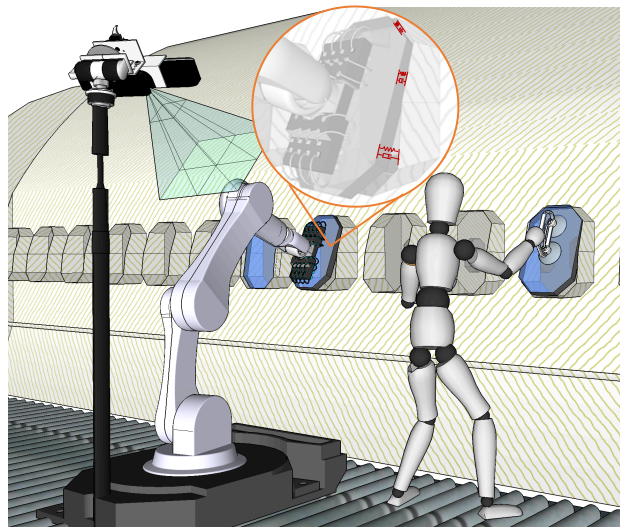
* Contributed equally to this work.

The authors are with the HRI² Lab, Istituto Italiano di Tecnologia, Genoa, Italy.

emir.mobedi@iit.it, nicola.villa@iit.it

Fig. 1. The proposed control framework aims to improve robot adaptation to environmental constrains and external forces in assembly tasks.

(RCC) mechanism are among the most effective yet low-cost techniques to this problem, which allow a robot to bear the lower amount of torques at about the orientation of the end-effector [6]. However, it is known that such passive solutions lack flexibility and scalability.

With the aim to improve the flexibility of robotic assembly, adaptive control techniques [7], such as hybrid position/force control algorithms have been proposed [8]–[10]. The general idea among the existing control techniques is to implement a position control along the lateral direction while pushing the robot towards the direction of the insertion. However, setting a desired force at the end-effector limits the application area of the robots, since force control is usually suitable for tasks in which all the constraints are predefined. In fact, sudden contact losses in direct force control approaches can result in unexpected robot behaviours, which may cause damage to human coworkers or the environment.

To overcome these limitations, the assembly task can be achieved using impedance control [5], by adjusting the control gains in each axis with respect to the geometrical constraints or friction (due to different clearance levels) [11]. Nevertheless, the choice of impedance gains should be specific to task conditions, which is usually chosen by robot programmers. This can severely reduce the flexibility of the robot to various interaction scenarios (e.g., see concept illustration in Fig. (1)).

As an alternative technique to the model-based approaches [7] described above, model-free algorithms based on machine learning have been proposed. These techniques can be di-

vided in two categories, i.e., *learn from demonstration* (LfD) and *reinforcement learning* (RL). The former one is implemented by recording human data during assembly tasks, which is followed by the programming of the robot [12]. Instead, the latter is based on processing the data acquired from the environment in several attempts of executing a task [13], [14]. The main drawback of such techniques is the need for large experimental data (LfD) or trials (RL) to ensure the execution of an assembly task subject to varying geometric constraints.

In response to the above mentioned limitations, we propose an adaptive impedance control strategy to regulate the level of robot adaptation to the environmental constraints and external forces in assembly tasks. Unlike the method of estimating the robot parameters via the gradient [15] or least square method [16], the proposed method modifies the control gain intuitively depending on the external forces, which is occurred in the physical interaction.

In the framework we present in this paper, the controller implements a stiff profile along the insertion direction, which is detected by an external vision system, and an adaptive impedance controller in all other axes. This consideration is to facilitate the insertion of the peg in the hole, since the stiff profile will help overcome the friction in the direction of insertion, while the 'compliant-as-needed' profile in other directions maximise the alignment of the peg with the hole during insertion. The later will eliminate the generation of unnecessarily high interaction forces in axes other than the insertion.

The efficiency of the proposed controller is evaluated experimentally in a peg-in-the-hole task, using a 7 degrees of freedom (DoF) Franka robot. We compare the proposed controller with two other control implementations, in which robot Cartesian stiffness is set to a stiff and a compliant value. The rest of the paper is structured as follows. In section II and II-D, the working principle of the controller, the details of the vision system, and the implemented Finite State Machine are explained. In section III, the experimental setup is introduced. In section IV, the experimental results are discussed. Finally, conclusion and the future studies are discussed in section V.

II. THE ROBOTIC ASSEMBLY FRAMEWORK

This section describes the individual components of the proposed framework, which consists of an adaptive controller, an external vision system, an interaction planner module, and a finite state machine (FSM).

A. Adaptive controller module

To characterize the adaptive controller, first a brief overview of the impedance control theory is provided. Defined the generalised coordinates as $q \in \mathbb{R}^n$, the dynamic equation of motion for n-DoFs robot can be written as:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau + \tau_{ext}, \quad (1)$$

where $M(q)$ is the $n \times n$ inertia matrix, $C(q, \dot{q})$ is the Coriolis/centrifugal matrix, $g(q)$ is the vector of gravity

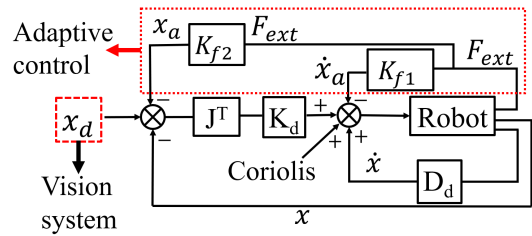Fig. 2. The block-diagram of the proposed control algorithm for robotic assembly.

joints torques, and τ is the vector of actuated joint torques. The robot's dynamic equation includes the reaction torques $\tau_{ext} = J(q)^T F_{ext}$, which are exploited by the k linearly independent constraint forces $F_{ext} \in \mathbb{R}^k$ and the Jacobian matrix $J(q) \in \mathbb{R}^{k \times n}$. For instance, F_{ext} is a external wrench at the robot's end-effector in Cartesian space (i.e. $k = 6$). Hence, the dynamic behavior in Cartesian space can be written as:

$$\Lambda_d \ddot{\tilde{x}} + D_d \dot{\tilde{x}} + K_d \tilde{x} = F_{ext}, \quad (2)$$

where $\tilde{x} = x - x_d \in \mathbb{R}^6$ represents the Cartesian error calculated with respect to the desired Cartesian coordinates x_d. In addition, Λ_d, D_d and K_d are the desired Cartesian ($\in \mathbb{R}^{6 \times 6}$) inertia, damping and stiffness matrices respectively, which are positive definite and symmetric. By avoiding the inertia shaping and assuming the desired velocity to be zero, (2) can be simplified to:

$$D_d \dot{\tilde{x}} + K_d \tilde{x} = F_{ext}. \quad (3)$$

Inserting (3) into (1) results in the target impedance behaviour [17]:

$$\tau = g(q) + C(q, \dot{q})\dot{q} + J(q)^T F_{ext}. \quad (4)$$

However, the control gains K_d and D_d of such a classical impedance controller require some appropriate values, which rely on the specific geometrical task constraints, hence, it will be severely limited to cope with uncertainties of a task (e.g. the mismatches between the estimated and actual observation). To resolve this issue, the adaptive controller employed in the framework enables a flexible behaviour according to the external wrench. This can be achieved by implementing the regulation term x_a ($x_{adjusted}$) in (3) as follows

$$D_d(\dot{\tilde{x}} - \dot{x}_a) + K_d(\tilde{x} - x_a) = F_{ext}, \quad (5)$$

where,

$$\dot{x}_a = K_{f1} F_{ext}, \quad x_a = K_{f2} F_{ext}, \quad (6)$$

K_{f1} and K_{f2} ($\in \mathbb{R}^{6 \times 6}$) are symmetric and positive definite matrices and defined as the inverse damping and stiffness matrices, respectively. Considering that the two matrices are normally set equal, in the following it is referred to these two as a single matrix K_f, called compliance matrix according to [18].

The overall block-diagram of the proposed adaptive controller is illustrated in Fig.2. If there is no external interaction, the adaptive controller (the upper part in Fig.2) does
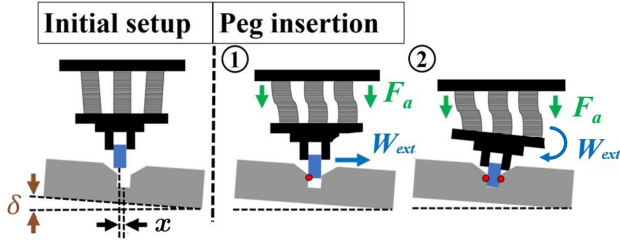
Fig. 3. Illustration of RCC system's behaviour. x and δ are the initial linear and rotational mismatch between peg and hole, respectively. The red circles are the Contact Points (CPs) between peg and hole, F_a is the applied force, while W_{ext} is the external wrench. In peg insertion phase, the resulting forces at CPs act: (1) pushing the RCC laterally with a single CP; (2) Rotating the RCC with two CPs.

not interfere with the controller input since $\boldsymbol{F}_{ext} = \boldsymbol{0}$, and behaves as the classical impedance controller with $\boldsymbol{K}_d$ and $\boldsymbol{D}_d$ positive definite control gains. On the contrary, if the external wrenches are applied, $\dot{\boldsymbol{x}}_a$ and $\boldsymbol{x}_a$, which are obtained by the direction and the amplitude of the external forces, regulate the target impedance behaviour through modification of the position/velocity error. As a consequence of this condition, the robot enables compliant motion in a certain direction where $\boldsymbol{K}_f$ is applied.

A high (low) value inside $\boldsymbol{K}_f$ leads to a high (low) correction to position/velocity error ($\tilde{\boldsymbol{x}}/\dot{\tilde{\boldsymbol{x}}}$) along different Cartesian coordinates; therefore, the robot responds with lower (higher) forces to the environment, showing a more compliant (stiffer) behavior. About $\boldsymbol{K}_d$ and $\boldsymbol{D}_d$ values in (5), these can be chosen high enough to ensure high tracking capability of manipulator in free motion. It is therefore clear how the controller behaves as a virtual and active RCC system (Fig.3), which actively modulates the response of a highly stiff robot in case of interaction with the workpiece.

In a peg-in-hole assembly as shown in Fig.3, for example, the external forces are applied to the peg around orientation axes, and the adaptive controller moves the peg along with the opposite orientation of the geometric constraint due to the adjusted trajectory.

Lastly, a stability analysis of the proposed adaptive controller is presented. The asymptotic stability of the classical impedance controller in (3) is considered via a Lyapunov function, as in [17], [19], [20]. Now the following Lyapunov function is proposed:

$$\boldsymbol{V}(\tilde{\boldsymbol{x}}, \dot{\tilde{\boldsymbol{x}}}, t) = \frac{1}{2}\dot{\tilde{\boldsymbol{x}}}^T \boldsymbol{M}(\boldsymbol{x})\dot{\tilde{\boldsymbol{x}}} + \frac{1}{2}\tilde{\boldsymbol{x}}^T \boldsymbol{K}_d \tilde{\boldsymbol{x}}. \qquad (7)$$

Considering the adaptive controller, the open version of (5) can be obtained by substituting (6) into (5):

$$\boldsymbol{D}_d(\dot{\tilde{\boldsymbol{x}}} - \boldsymbol{K}_{f1}\boldsymbol{F}_{ext}) + \boldsymbol{K}_d(\tilde{\boldsymbol{x}} - \boldsymbol{K}_{f2}\boldsymbol{F}_{ext}) = \boldsymbol{F}_{ext}, \qquad (8)$$

By isolating $\boldsymbol{F}_{ext}$ in (8):

$$\underbrace{\frac{\boldsymbol{D}_d}{1 + \boldsymbol{K}_{f1}\boldsymbol{D}_d + \boldsymbol{K}_{f2}\boldsymbol{K}_d}\dot{\tilde{\boldsymbol{x}}} + \overbrace{\frac{\boldsymbol{K}_d}{1 + \boldsymbol{K}_{f1}\boldsymbol{D}_d + \boldsymbol{K}_{f2}\boldsymbol{K}_d}}^{\boldsymbol{K}^*}\tilde{\boldsymbol{x}} = \boldsymbol{F}_{ext}}_{} \qquad (9)$$

It is clear that $\boldsymbol{K}_d$ in (3) is replaced with $\boldsymbol{K}^*$, which consists in a division between positive definite matrices.

Therefore, recalling (7), this concludes that the controller is asymptotically stable.

B. Vision system module

This section describes the vision module, responsible for the object's pose detection in the robot workspace. Thanks to this module, the robot is able to understand the task's environment that includes the target points. In this work, we target a *peg in a hole* task. Moreover, the vision system is in charge of identifying the insertion axis, that is crucial for assembly tasks. The required reference frames are defined as: base frame (fixed) Σ_B, tool frame Σ_H, end-effector frame Σ_{EE} and camera frame Σ_C. The location of these references will be shown in the following section.

To detect position and orientation of an assembly part (here, a hole), we implemented a template matching technique [21]. This approach ensures good approximation of object pose in the work space, through the fitting on scene point cloud of an object's model. The algorithm operates on point clouds, which contain position information of points in the space, without neither the need of further conversions from pixels to meters space nor issues related to work space illumination, as in the case of image based algorithms. Through this approach, it is possible to select a point on the template cloud which can be used for the detection of the hole fixed frame. In case of peg in a hole, the bottom of the hole model point cloud was selected. In the following, we refer to this point as P_H.

Once the model is fitted on the environment point cloud, a rigid transformation is applied from the new reference frame (rotated according to the fitted template) to P_H. The output of the algorithm is a transformation matrix $\boldsymbol{U}_H^C \in \mathbb{R}^{4\times4}$ from the camera frame (Σ_C) to a frame fixed to the cylindrical hole (Σ_H), translated according to P_H and rotated according to the hole inclination. The z axis of the new frame is set along the main axis of insertion, i.e. the cylinder axis. The transformation matrix $\boldsymbol{U}_H^B$ from hole to robot base frame (Σ_B) is obtained through a multiplication between the object pose in camera frame $\boldsymbol{U}_H^C$ and the camera pose in robot base frame $\boldsymbol{U}_C^B$ by $\boldsymbol{U}_H^B = \boldsymbol{U}_C^B \ \boldsymbol{U}_H^C$.

C. Interaction planner module

Interaction planning module consists in the generation of the robot trajectories to reach, roughly align the peg with the hole and insert the peg, as well as the planning of the robot Cartesian stiffness to ensure an RCC-like behaviour at robot end-effector. The first is obtained by means of a trajectory planner implementing a 5th order polynomial trajectory that enables smooth trajectory planning between the current and the target pose of the robot end-effector. The target is expressed through a transformation matrix $\boldsymbol{T}_{target}^B$ from base frame (Σ_B) to target frame. The second is achieved via the modification of the original compliance matrix $\boldsymbol{K}_f$, characterized relatively to the hole fixed frame. In case the estimated hole frame is not aligned with the robot base frame (e.g. the hole is inclined), $\boldsymbol{K}_f$ is transformed into the base reference according to the hole orientation. Hence, the
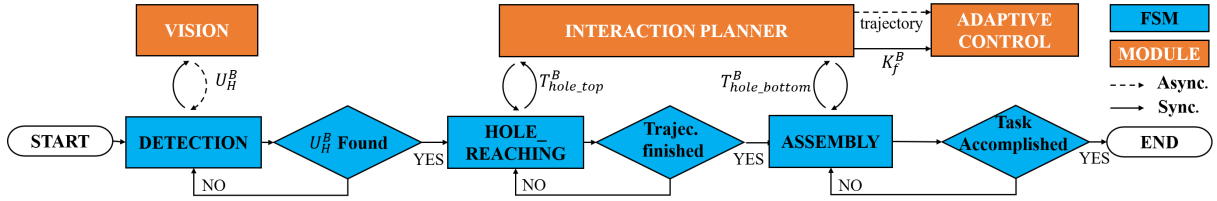
Fig. 4. Finite State Machine and Modules communication chart

transformed K_f can be employed in the adaptive controller with $\tilde{x}$ and F_{ext} in (5). For sake of clarity, the compliance matrix with respect to the hole fixed frame is called K_f^H, while K_f^B corresponds to the compliance matrix in robot base frame. To perform this transformation, the rotational part of matrix U_H^B (explained in sec.II-B) is considered, i.e. $R_H^B \in \mathbb{R}^{3\times3}$. The matrix $\Re \in \mathbb{R}^{6\times6}$ is built as:

$$\Re = \begin{bmatrix} R_H^B & 0_{3\times3} \\ 0_{3\times3} & R_H^B \end{bmatrix}, \qquad (10)$$

K_f^B is computed by $\Re K_f^H \Re^T$ which represents the Singular Value Decomposition of the symmetric stiffness matrix K_f^B. In such a way, the interaction planner can assign high stiffness value on the major axis (insertion axis) and the adaptive behavior in (5) along other DoFs.

D. Finite State Machine

To make the above modules communicate, a finite state machine (FSM) is implemented. The FSM consists in 3 states: *detection* (S1), *hole_reaching* (S2), *assembly* (S3).

The unified control framework is illustrated in Fig.4. The control flow shows that S1 triggers the vision module, asking for hole detection. This module returns the transformation matrix U_H^B. If the hole is found, FSM passes to S2, which triggers the trajectory planner to compute a path to the hole ($T_{hole\_top}^B$); the module passes the path to the controller as a sequence of equilibrium points, together with K_f^B, which is computed only once according to hole pose. If the reaching is successful, FSM triggers S3 which sends a request to trajectory planner, i.e. going down to $T_{hole\_bottom}^B$ for inserting the peg. If the task is accomplished, the FSM stops.

III. EXPERIMENTAL SETUP

The experimental setup and reference frames are illustrated in Fig.5. The experimental evaluation involved a peg-in-hole task. The first set of experiments was named Experiment A, and designed to compare the performance of the adaptive controller with the classical impedance controller. Following that, the performance of the complete framework (Experiment B) was evaluated by a realistic assembly in the inclined hole setup, where was demonstrated the integration of the modules. The hardware setup included the Franka Emika's Panda robot equipped with an ASUS Xtion Pro Live RGB-D camera. A hole with $34.6\,\mathrm{mm}$, and three pegs with $32.6\,\mathrm{mm}$, $33.6\,\mathrm{mm}$, $34.4\,\mathrm{mm}$ diameters (difference clearances) were 3D printed for the experiments. Noteworthy, F_{ext} using in (4) were obtained from the Franka Emika's

external torque estimation algorithm [22], and represented in respect of Σ_B.

In Experiment A, the repeatability performance is evaluated. In the first type, named as the *stiff controller*, the Cartesian diagonal stiffness matrix had fixed-high values as $(K_x = K_y = K_z = 1000\,\mathrm{N/m}, K_{roll} = K_{pitch} = K_{yaw} = 30\,\mathrm{Nm/rad})$. Similarly, the *compliant controller* implemented fixed-low diagonal values $(K_x = K_y = 500, K_z = 1000\,\mathrm{N/m}, K_{roll} = K_{pitch} = K_{yaw} = 20\,\mathrm{Nm/rad})$. Finally, the *adaptive controller* was implemented as described in Section II-A. In this controller, the adaptive gains of K_f were selected as $(K_{fx} = K_{fy} = 0.001\,\mathrm{m/N}, K_{fz} = 0, K_{froll} = K_{fpitch} = K_{fyaw} = 0.14\,\mathrm{rad/Nm})$ for the first and the second peg. However, for the third peg, the gains were modified as $(K_{fx} = K_{fy} = 0.001\,\mathrm{m/N}, K_{fz} = 0, K_{froll} = K_{fpitch} = K_{fyaw} = 0.15\,\mathrm{rad/Nm})$ since the third peg has the lowest clearance among the others. During the experiments, K_{f1} and K_{f2} were assigned as the same value. $K_{fz} = 0$ intends to implement a fixed stiff behaviour in insertion direction (z). As mentioned in the controller section before, the selection of K_f depends on the geometrical constraints in the robot environment. If the higher forces are expected in a direction, the K_f should be assigned with higher values along that direction to decrease the forces. In our application, since the goal is to be adaptive around the roll, pitch and yaw to allow the peg insert the hole, the selection of K_{froll}, K_{fpitch} and K_{fyaw} were kept higher with respect to x, y and z directions.

x, y positions and pitch angles of the robot (in Σ_B) were kept the same for overall trials. Either α or β orientations was varied to check whether RCC can compensate those
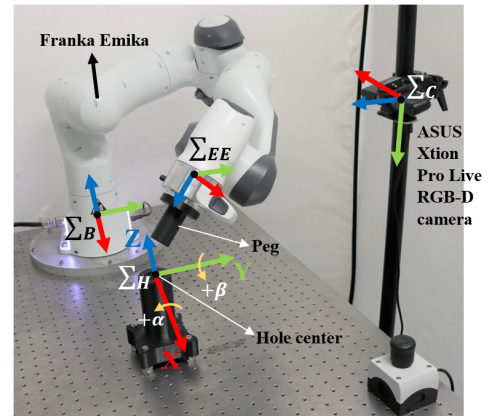

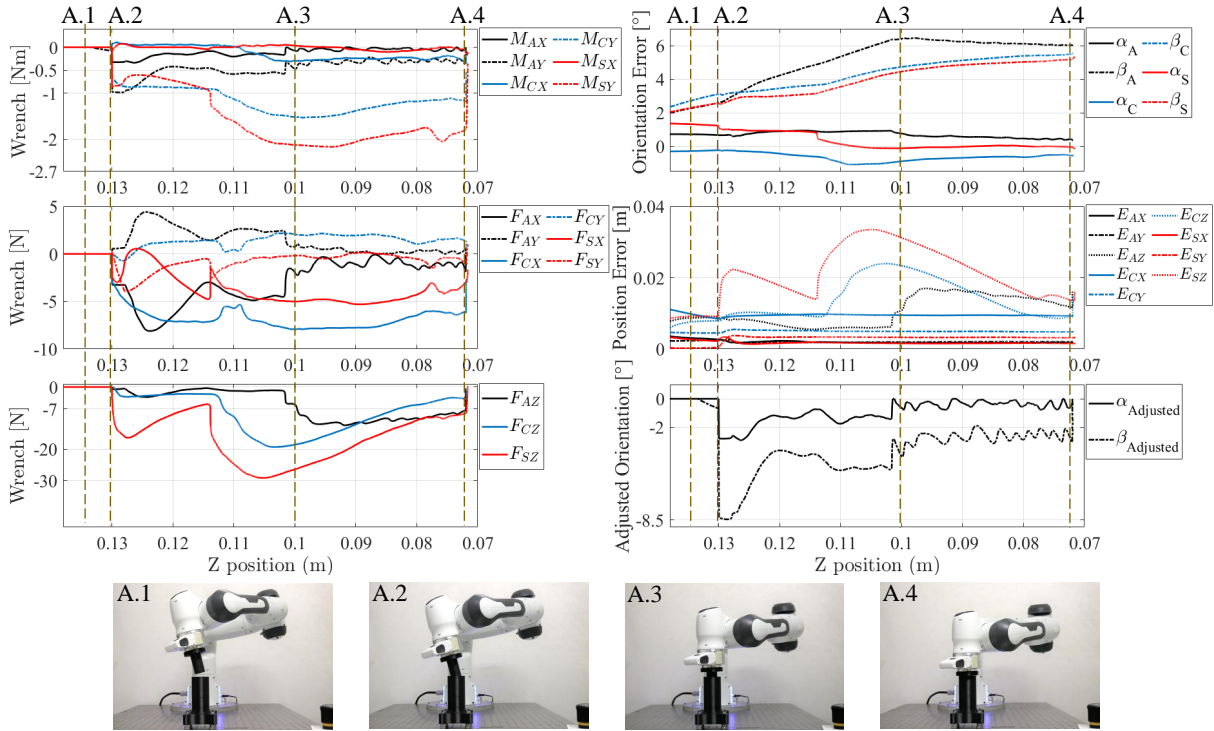Fig. 5. Illustration of the experimental setup.

Fig. 6. Snapshots of the repeatability test for 3rd peg at $\beta = -0.1$ rad. In the plots, the colours and sub characters are used to represent the three controllers. Black: Adaptive (A), Red : Compliant (C), Blue : Stiff (S). A.1) Initial phase of the peg, A.2) The beginning phase of the interaction, A.3) The half-inserted phase, A.4) The end of the insertion.

mismatches. In the following steps, the procedure is written for one pose (α varied, β constant).

1) Go to the initial pose ($z = 0.13$ m : α_1 : β_1).
2) Go to the final pose ($z = 0.07$ m : α_1 : β_1).
3) If the peg half-inserted or stucked, record the wrench and write "−" for the next poses (see Table(I)).
4) If the insertion is done ($z = 0.07$ m), send zero torque to the robot and record the wrench.
5) Repeat the abovementioned steps three times, and calculate the mean value and the standard deviation of wrench (z).
6) Go to the next test pose ($z = 0.13$ m : α_2 : β_1).

After that, the proposed experiment for the whole framework validation is performed with the hardware of Fig.1 but with an hole inclination of $\beta = 0.78$ rad. The controller parameters were kept the same with the adaptive one for first peg. In Experiment B, the robot is first set to a random pose; then, it is required to reach the hole, detected by vision module (i.e. a relative position of 10 cm along z axis of Σ_H), and then to insert the peg (i.e. reaching a relative position of 1 cm along the same direction). To ensure the insertion, an additional rotation around hole axes is added to the target, i.e. $\alpha = -0.25$ rad and $\beta = 0.25$ rad.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

A. Controller Validation

In Fig.6, six plots are reported for one of the repeatability tests poses to analyze the generalized interaction forces, and position and orientation error. Also, x_a is illustrated for roll ($\alpha_{adjusted}$) and pitch ($\beta_{adjusted}$) axis.

At phase 1 (A.1), the orientation errors of the compliant controller are greater than the stiff and adaptive ones because of its gains. In the meantime, the adjusted orientation is zero as there is no interaction. At phase 2 (A.2), the values of $\alpha_{adjusted}$ and $\beta_{adjusted}$ increase in the negative direction to reduce the position and velocity error (see (5)). This adaptation achieves forces and torques that are significantly lower for the adaptive controller in constraint axes (and not the insertion). To clarify, for the wrench along x and y directions, the data for compliant and adaptive controllers are relatively closer to each other than the force data in (z) since $\boldsymbol{K}_f$ is selected as lower ($K_{fx} = K_{fy} = 0.001$). At the next phase (A.3), the adaptive controller fully compensated for the hole constraints, which could be easily seen from the orientation error graph (β_A). Finally, the insertion was done, and the desired wrench was set to zero.

In Table I, the results of 51 trials (17 poses for three pegs, pointed as P-1, P-2, P-3) are reported. The success rate is computed according to the trial number. When α is 0.34 rad for P-3, the adaptive controller can achieve the task. However, the opposite pose ($\alpha = -0.34$ rad) cannot be accomplished. The reason arises from the force estimation error of Franka Emika (the error varies from axis to axis), leading to acquire different wrench according to (6).

When α or β test poses are greater than $|0.24|$ rad, both compliant and stiff controller are not capable of achieving the task, which concludes that the proposed controller bears the highest success rate among others.

TABLE I

Repeatability Test Results. The value is reported as: mean (standard deviation).

Poses [rad]	Adaptive [N]			Compliant [N]			Stiff [N]		
	P-1	P-2	P-3	P-1	P-2	P-3	P-1	P-2	P-3
0	-8.1(1.2)†	-14.6(4.8)†	-14.2(5.3)†	-4.8(1.2)†	-2.7(0.9)†	-14.4(1.9)†	-3.1(1.3)†	-2.5(0.4)†	-15.0(0.9)†
α=0.14	-7.3(0.7)†	-15.5(0.6)†	-20.1(1.1)†	-18.6(1.8)†	-27.3(0.9)§	-43.7(1.6)§	-18.9(1.3)††	-30.0(1.5)§	-45.1(0.8)§
α=0.24	-13(1.0)†	-21.2(0.6)†	-34.8(5.9)†	-39.1(1.4)††	–	–	–	–	–
α=0.34	-18.6(0.4)†	-23.1(0.1)†	-36.6(2.6)†	–	–	–	–	–	–
α=0.44	-27.7(1.1)†	-33.7(1.1)†	-48.83(2.4)§	–	–	–	–	–	–
α=-0.14	-10.4(0.4)†	-12.2(0.2)†	-7.2(0.5)†	-17.5(0.4)†	-17.8(3.0)††	-49.4(0.3)§	-20.7(0.6)†	-24.7(1.7)††	-47(1.6)§
α=-0.24	-14.1(0.3)†	-20.1(1.6)†	-41.1(1.1)†	-37.1(0.9)††	–	–	-36.6(2.4)††	–	–
α=-0.34	-23.2(0.7)†	-29.7(0.8)†	-50.7(0.5)††	–	–	–	–	–	–
α=-0.44	-33.5(2.6)†	-42.5(0.8)†	–	–	–	–	–	–	–
β=0.1	-18.2(2.1)†	-15.3(0.6)†	-33.8(0.9)†	-17.9(1.2)†	-20.3(3.8)††	-46.4(2)§	-18.5(1.6)††	-28.4(3.3)§	-47.2(0.5)§
β=0.2	-24.6(0.9)†	-26.7(1.4)†	-50.1(2.0)†	-36.5(0.3)††	–	–	–	–	–
β=0.3	-29.3(0.1)†	-33.5(4.7)†	-50.7(1.4)†	–	–	–	–	–	–
β=0.4	-36.8(2.1)†	-39.5(2.2)†	-48.3(1.2)§	–	–	–	–	–	–
β=-0.1	-5.8(3.1)†	-12.9(1.6)†	-24.4(5.7)†	-8.6(1.6)†	-10.5(1.7)†	-21.1(6.5)†	-9.7(1.3)†	-15.5††(3.4)	-20.3(5.6)†
β=-0.2	-6.7(1.2)†	-13.2(1.0)†	-23.6(2.4)†	-24.8(1.4)††	-40.9(4.1)††	-41.7(1.5)§	-34.4(3.0)††	–	-47.3(1.2)§
β=-0.3	-12.7(2.2)†	-25.3(3.4)†	-38.2(5.5)†	–	–	–	–	–	–
β=-0.4	-23.9(3.7)†	-28.4(1.7)†	-45.3(2.4)§	–	–	–	–	–	–
Success rate	**91.00%**			**17.64%**			**11.76%**		

† Inserted, †† half inserted, § stucked

B. Framework Validation

In this section, the validation results for the whole framework are presented. Figure 7(a) shows experiment's phases, i.e. *hole detection* (B.1), *hole entrance reaching* (B.2), *insertion phase* (B.3) and *final phase* (B.4). Figure 7(b) illustrates the results in terms of pose error and interaction wrenches, both expressed in base frame.

Here, the experiment is explained in details. Once the hole is detected (B.1), the robot starts moving toward a target (B.2), i.e. the hole entrance. Vision output is shown in enlargement 1 of Fig.7(a), where the red point cloud corresponds to the hole template matched on the scene. As in 1st and 2nd row of Fig.7(b), during B.2 the control behaves as an impedance controller with high stiffness parameters, tracking perfectly the desired trajectory with a maximum position and orientation error of $1\,\mathrm{cm}$ and $0.05\,\mathrm{rad}$, respectively. Indeed, the wrenches are all zero during this phase and consequently $\boldsymbol{x}_a$ and $\dot{\boldsymbol{x}}_a$ in (5).

On the contrary, during B.3, the wrenches start to increase. The enlargement 2 of Fig.7(a) shows how this is caused by a linear and rotational misalignment, which occurs between peg and hole due to some error in detection. In this situation, the robot keeps on pushing the peg down, due to the predefined parameter inside $\boldsymbol{K}_f$, leading the perceived wrenches to increase. The presence of external forces causes an increment in robot compliance along DoFs with non-zero $\boldsymbol{K}_f$ parameters. The combined action of the z-axis force and the increasing compliance leads the robot to correct the mismatch and accomplish the task inserting the peg (B.4). The increment in position and orientation errors during B.3 is caused by the higher compliance of the robot in this phase which allows a higher deviation from the target trajectory according to environment constraints. The manipulator is therefore guided by the environment (here the hole borders) in correcting the pose for accomplishing the task.

During B.4, the wrenches and the error drop down. However, the latter remains constant to a non-zero value for x, z and orientation coordinates. The final errors in orientation are caused by the mismatch correction. The final errors along x and z are due to both mismatch correction and unreachability of the target. Indeed, the peg is completely inserted and stops before it reaches the target required for the insertion, i.e the hole bottom. Nevertheless, the task is accomplished and the controller is stopped.

V. CONCLUSION

In this work, we proposed a novel adaptive framework for accomplishing robotic assembly tasks. The framework is composed of an adaptive controller, a vision system, and a trajectory planner, each communicating through a Finite State Machine. The controller was validated through experiments on a typical assembly action, i.e. *peg-in-hole*, as a representative example of tasks with dynamic uncertainties. First, the adaptive controller was compared with two classical impedance controller to demonstrate the strength of the proposed strategy. Next, the integrated framework was tested: the vision system was able to detect an inclined hole on the work table and the robot was able to reach the hole with very low pose error during the tracking phase. The adaptive controller enabled the robot to correct some linear and rotational misalignment between peg and hole, to accomplish the insertion. Results demonstrated the high potential of the proposed approach in increasing the flexibility of robotic assembly operations.

REFERENCES

[1] S. Wolf, G. Grioli, O. Eiberger, W. Friedl, M. Grebenstein, H. Höppner, E. Burdet, D. G. Caldwell, R. Carloni, M. G. Catalano,
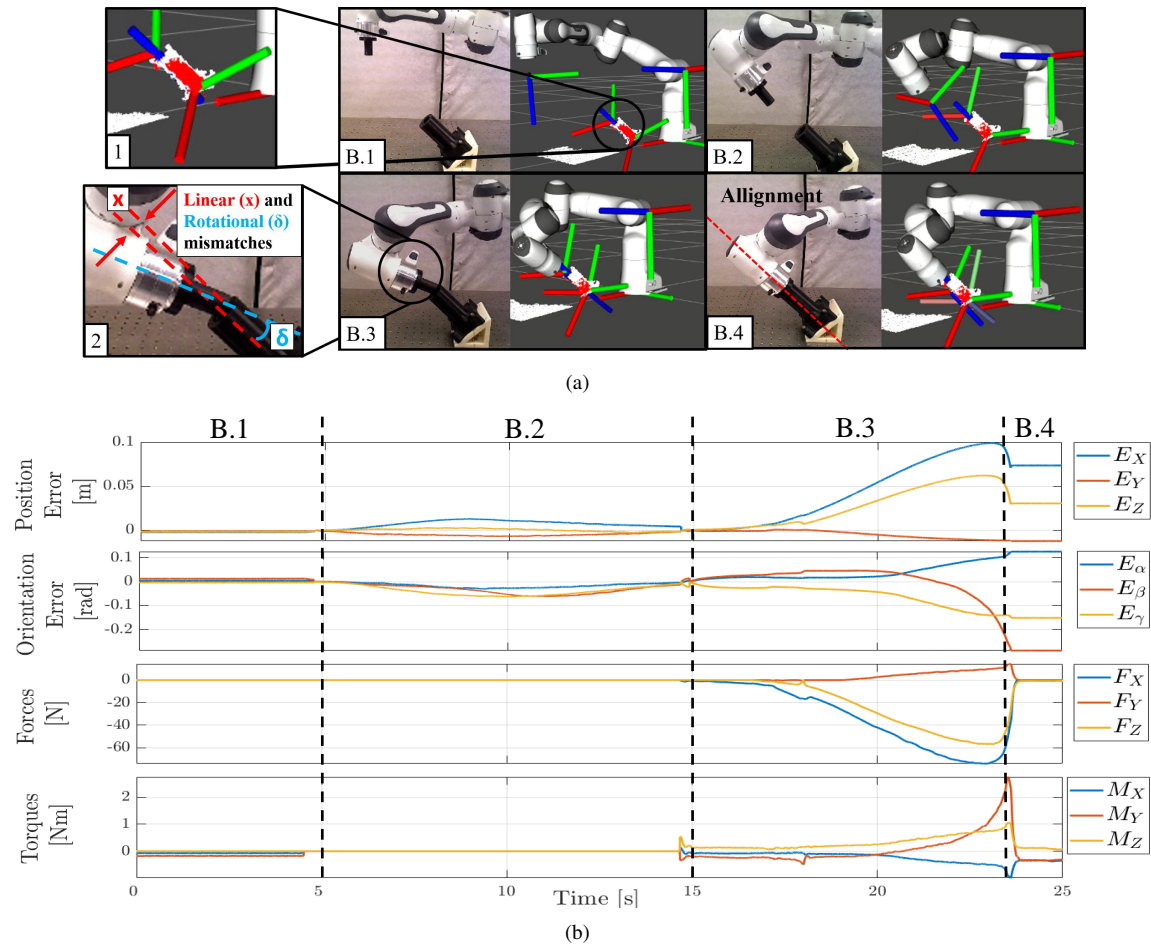
Fig. 7. Framework validation in terms of the sequence of experimental phases (top) and the experimental results for pose error and wrenches (down)

D. Lefeber, S. Stramigioli, N. Tsagarakis, M. Van Damme, R. Van Ham, B. Vanderborght, L. C. Visser, A. Bicchi, and A. Albu-Schäffer, "Variable stiffness actuators: Review on design and components," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 5, pp. 2418–2430, 2016.

[2] L. Peternel, N. Tsagarakis, and A. Ajoudani, "A human–robot co-manipulation approach based on human sensorimotor information," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 7, pp. 811–822, 2017.

[3] J. K. Salisbury, "Active stiffness control of a manipulator in cartesian coordinates," pp. 95–100, 1980.

[4] D. E. Whitney, "Historical perspective and state of the art in robot force control," *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, vol. 2, pp. 262–268, 1985.

[5] R. Ahmad and P. Plapper, "Safe and automated assembly process using vision assisted robot manipulator," *Procedia CIRP*, vol. 41, pp. 771 – 776, 2016.

[6] N. Ciblak and H. Lipkin, "Design and analysis of remote center of compliance structures," *J. Field Robotics*, vol. 20, pp. 415–427, 2003.

[7] J. Xu, Z. Hou, Z. Liu, and H. Qiao, "Compare contact model-based control and contact model-free learning: A survey of robotic peg-in-hole assembly strategies," 04 2019.

[8] H. Abdullah, Mustafa W.and Roth, M. Weyrich, and J. Wahrburg, "An approach for peg-in-hole assembling using intuitive search algorithm based on human behavior and carried by sensors guided industrial robot," *International Federation of Automatic Control*, vol. 48, pp. 1476–1481, 2015.

[9] H. Park, J. Park, D.-H. Lee, J.-H. Park, M.-H. Baeg, and J.-H. Bae, "Compliance-based robotic peg-in-hole assembly strategy without force feedback," *IEEE Transactions on Industrial Electronics*, vol. 64, pp. 6299–6309, 2017.

[10] K. Zhang, M. Shi, J. Xu, F. Liu, and K. Chen, "Force control for a rigid dual peg-in-hole assembly," *Assembly Automation*, vol. 37, pp. 200–207, 04 2017.

[11] A. Ajoudani, N. Tsagarakis, and A. Bicchi, "Tele-impedance: Teleoperation with impedance regulation using a body–machine interface," *The International Journal of Robotics Research*, vol. 31, no. 13, pp. 1642–1656, 2012.

[12] B. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," pp. 469–483, 09 2017.

[13] I. Tadanobu, D. G. Magistris, A. Munawar, T. Yokoya, and R. Tachibana, "Deep reinforcement learning for high precision assembly tasks," pp. 819–825, 09 2017.

[14] J. Luo, E. Solowjow, C. Wen, A. J. Ojea, and A. M. Agogino, "A survey of robot learning from demonstration," pp. 2062–2069, 10 2018.

[15] S. LUO, S. AHMED, and M. ZRIBI, "Adaptive control of kinematically redundant robots," *IMA Journal of Mathematical Control and Information*, vol. 14, no. 3, pp. 225–253, 1997.

[16] E. Tatlicioglu, D. Braganza, T. C. Burg, and D. M. Dawson, "Adaptive control of redundant robot manipulators with sub-task objectives," in *2008 American Control Conference*, 2008, pp. 856–861.

[17] C. Ott, *Cartesian Impedance Control of Redundant and Flexible-Joint Robots*. Springer, 01 2008, vol. 49.

[18] H. H. Asada, "Lecture notes in introduction to robotics," 2005.

[19] V. Santibáñez and R. Kelly, "Strict lyapunov functions for control of robot manipulators," *Automatica*, vol. 33, no. 4, pp. 675 – 682, 1997.

[20] B. Paden and R. Panja, "Globally asymptotically stable 'PD+' controller for robot manipulators," *International Journal of Control*, vol. 47, no. 6, pp. 1697–1712, 1988.

[21] S. Hinterstoisserl, S. Holzerl, C. Cagniart, S. Ilic, K. Konolige, N. Navab, and V. Lepetit, "Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes," *IEEE International Conference on Computer Vision*, pp. 858–859, 2011.

[22] A. D. Luca, A. Albu-Schaffer, S. Haddadin, and G. Hirzinger, "Collision detection and safe reaction with the DLR-III lightweight manipulator arm," pp. 1623–1630, 2006.