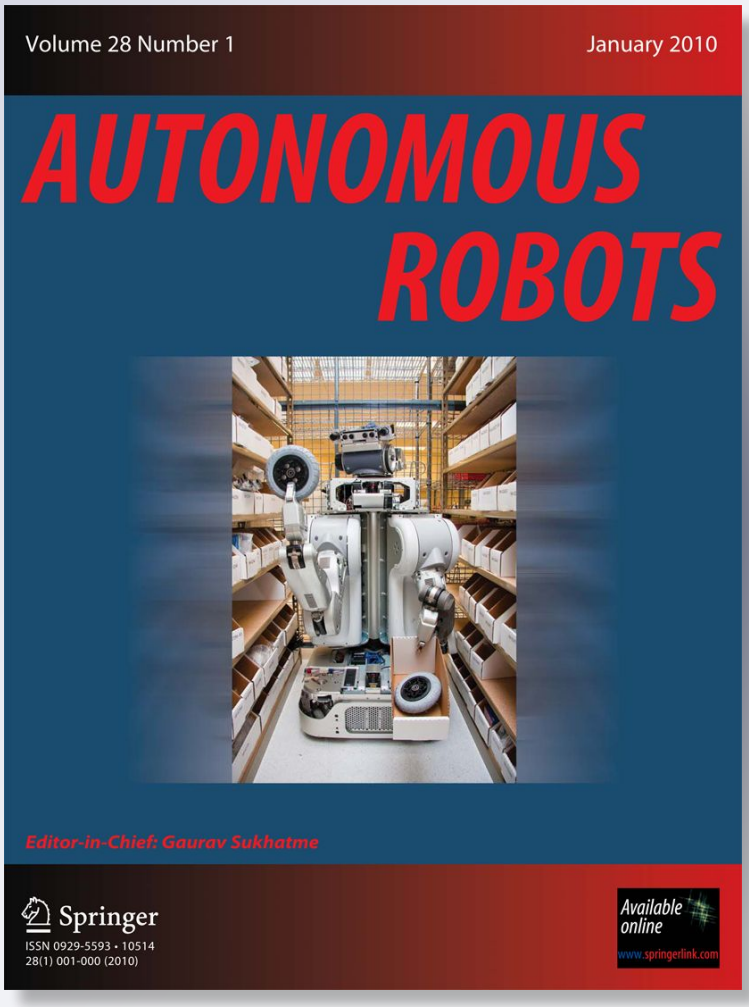
Analysis of autonomous cooperative assembly using coordination schemes by heterogeneous robots using a control basis approach

J. Rojas & Richard A. Peters

Autonomous Robots

ISSN 0929-5593
Volume 32
Number 4

Auton Robot (2012) 32:369-383
DOI 10.1007/s10514-012-9274-3



Springer

Auton Robot (2012) 32:369–383
DOI 10.1007/s10514-012-9274-3

Analysis of autonomous cooperative assembly using coordination schemes by heterogeneous robots using a control basis approach

J. Rojas · Richard A. Peters II

Received: 27 May 2011 / Accepted: 14 January 2012 / Published online: 2 February 2012

Abstract Robotic technology is quickly evolving allowing robots to perform more complex tasks in less structured environments with more flexibility and autonomy. Heterogeneous multi-robot teams are more common as the specialized abilities of individual robots are used in concert to achieve tasks more effectively and efficiently. An important area of research is the use of robot teams to perform modular assemblies. To this end, this paper analyzed the relative performance of two robots with different morphologies and attributes in performing an assembly task autonomously under different coordination schemes using force sensing through a control basis approach. A rigid, point-to-point manipulator and a dual-armed pneumatically actuated humanoid robot performed the assembly of parts under a traditional "push-hold" coordination scheme and a human-mimicked "push-push" scheme. The study revealed that the scheme with higher level of cooperation—the "push-push" scheme—performed assemblies faster and more reliably, lowering the likelihood of stiction phenomena, jamming, and wedging. The study also revealed that in "push-hold" schemes industrial robots are better pushers and compliant robots are better holders. The results of our study affirm the use of heterogeneous robots to perform hard-to-do assemblies and also encourage humans to function as holder's when working in concert with a robot assistant for insertion tasks.

Keywords Cooperative robots · Heterogeneous robots · Autonomous assembly · Force sensing · Force control

1 Introduction

Robotics technology is quickly evolving and demanding robots to perform more complex actions, with greater flexibility, autonomy, robustness, and in less structured environments (Brogardh 2007). Human-robot and multi-robotic teams are also becoming more frequent (Kruger et al. 2009; Shah et al. 2007; Sellner et al. 2005). Using cooperative teams of robots to perform a task is increasingly common in numerous applications to increase the task completion flexibility, reduce time-to-completion, and allow higher load capacities (Brogardh 2007). In particular, heterogeneous robot teams offer a wider range of abilities compared to homogeneous teams (Rehnmark et al. 2005).

The missions to the Moon currently planned by NASA requires the deployment of robots prior to the arrival of astronauts to prepare human life support sites. Part of the robot's work will be the assembly of modular structures such as solar arrays, antennas, and habitation modules under a certain degree of independence. Although the robot teams will be tasked and supervised by human operators on Earth it will be necessary for the robots to be autonomous over limited time intervals due to space communication latencies. Even without a delay, such minute teleoperation is difficult and prone to errors, specially if a number of operators were to jointly handle the mating of parts. The deployment of heterogeneous robot teams to perform assembly

Electronic supplementary material The online version of this article (doi:10.1007/s10514-012-9274-3) contains supplementary material, which is available to authorized users.

J. Rojas (✉)
Center for Intelligent Systems, Vanderbilt University School of Engineering, Nashville, TN 37235, USA
e-mail: rojas70@gmail.com

R.A. Peters II
Universal Robotics, Smith Springs Rd., Nashville, TN 37235, USA
e-mail: rap2@Vuse.Vanderbilt.Edu

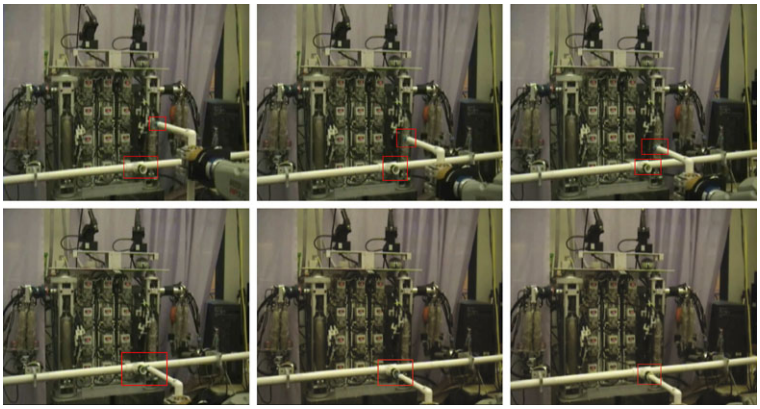


Fig. 1 Two heterogeneous robots perform a push-hold assembly

tasks autonomously in a dynamic environment will require that robot coordination use force sensing to drive adaptive and robust control strategies.

While there is an growing body of literature on heterogeneous robot teams there is no analysis on which robots work better for which tasks and under which roles. Nor, which tasks work better under different circumstances. Traditional assembly, which has been studied extensively for the last four decades (Rojas 2009), both under single-manipulator and multi-manipulator configurations (Caccavale et al. 2001); have been conducted in what the authors describe as a push-hold scheme. That is, one robot or manipulator actively pushes a mating part, while another passively holds the part. Human-robot and multi-robotic teams applied to novel applications like space construction (Ueno et al. 2003; Doggett 2002) and robot assistants (Shah et al. 2007) provide an opportunity for a non-traditional approach to assembly. This approach requires that two agents simultaneously push their parts to accomplish their mating. We call this a "push-push" scheme. This approach is motivated in Ozaki et al. (2004) to fit modular micro-satellite parts. It is also biologically inspired in the way humans perform hard-to-do assemblies, whereby humans empirically resort to this approach if connecting two parts is difficult.

Our work focuses on the first step of the construction problem, that is the insertion of parts across two robots. It does not address how to build more parts, nor whether it is done on-orbit, or on-surface. Several works describe further steps of the construction problem in different environments (Doggett 2002; Sellner et al. 2005; Simmons et al. 2007, and Huntsberger et al. 2004). Doggett's approach is an on-orbit laboratory assembly system that connected 102 truss elements; Sellner et al. and Simmon et al. use three robots with specialized skills to locate, carry, and manipulate truss structures for assembly. Huntsberger et al. use two mobile robots with manipulators to transport and assemble stanchions.

While difficult-to-do assemblies involve higher contact forces, humans still use 'push-push' approaches to execute the insertions. Traditional assemblies have avoided this approach for the same reason and have sought to use compliance, often in the form an end-effector, to reduce potentially high and damaging contact forces present in assemblies using rigid manipulators (Pratt and Williamson 1995).

This paper studied performance differences between traditional "push-hold" assemblies and "push-push" schemes carried out by two robots with very different morphologies and attributes: a rigid manipulator and a compliant humanoid robot. The paper also studied which robot was better suited for push or hold roles in traditional "push-hold" assemblies by analyzing their force and moment responses. The study provides valuable insight to understand the relative advantages of using robots of particular morphologies and attributes in a team of heterogeneous robots for assembly tasks.

To this end, the authors tasked an anthropomorphic dual-armed, six DoF, pneumatically actuated robot ISAC (Rojas and Peters II 2005) and a rigid, point-to-point, six DoF industrial manipulator HP3JC robot (Rojas and Peters II 2009) to perform a collaborative assembly task under two coordination schemes: the "push-hold" scheme and the "push-push" scheme. The assembly task had the HP3JC hold a male truss with a barret hand while the dual-armed humanoid used a PVC T-connector construct to hold a female truss using both arms. For the "push-hold" scheme two experiments were performed to test each robot under a different role. In the first experiment, the HP3JC drives the insertion with the male truss while ISAC holds, and in the second experiment ISAC drives the insertion with the female truss while the HP3JC robot holds the male truss. For the "push-push" scheme an experiment is performed where both robots drive the insertion of parts. Screen-shots of the push-push scheme are shown in Fig. 1. An in-house built distributed multi-agent architecture known as the Intelligent Machine Architecture was used in concert with a control basis framework—a modular control basis architecture—to facilitate the deployment of the robotic testbed while achieving a robust and flexible control paradigm (Rojas and Peters II 2011b). The control basis is a practical framework

for robotics development and has been used to implement bipedal walking, manipulation, grasping (Brock et al. 2005), mobile-agent flocking (Antonelli et al. 2010), and traditional assemblies (Rojas and Peters II 2011b). The control basis decomposes a control problem into a series of compound modules that are sequenced concurrently as part of an instruction set of optimized objectives. Such recasting allows for an autonomous, robust, and flexible formulation and extension of a control solution (Brock et al. 2005).

The study found that the "push-push scheme" completed insertion tasks faster than the "push-hold" scheme and reduced the likelihood of stiction, jamming, and wedging. Finally, the study also presents force-moment signature patterns of assembly phenomena for coupled rigid-compliant robot teams which are valuable in understanding the challenges of coupling robots of very different attributes and can help design frameworks to synthesize and evaluate tasks across teams of heterogenous robots. The study also found that for the "push-hold" scheme, having the industrial robot be the pusher and the compliant robot be the holder was more desirable due to resulting smoother insertions with lower contact forces compared to the assembly with reversed roles.

2 The control basis framework

A control basis decomposes a complex control system into a set of modular control elements that when connected appropriately synthesize a variety of behaviors. A control basis consists of any number of closed loop controllers derived from a set of control laws. As asserted by Huber (2000), the control laws are designed to yield asymptotically stable and predictable behavior for different robots. Each control law discretizes the continuous space into discrete basins of attraction and can compensate for a limited range of perturbations and uncertainties while still converging to the attractor. The control basis approach uses nullspace projections to optimize multiple control goals simultaneously and mitigate singularities associated with the projection of space actions onto other spaces. The control composition is also implemented through the selection of relevant sensor and actuator resources to produce flexible structural solutions beyond modular nullspace behavioral control (Brock et al. 2005). In effect, the approach allows control elements to be re-used and generalized to different solutions depending on the context (Brock et al. 2005).

2.1 Mathematical derivation

Primitive controllers ϕ_i, where $i = 1 \sim n$, form basis controllers from a set of controllers, Φ, such that $\phi_i \in \Phi$. A primitive controller optimizes a partitioned portion of a designated control space and can be understood as the minimization of a discrete basin of attraction. The basins of attraction are formulated through artificial potential functions defined over a typed domain, X_i, and are defined as the square of the error: $\phi_i(\rho) = \rho^T \rho$, where ρ is the difference between the reference input and the plant input, $\rho = \mathbf{q}_{ref} - \mathbf{q}$, at every time step.

Each controller reaches its objective by performing greedy descent, $\nabla\phi_i$, on the artificial potential function, while engaging sensor and motor resources. The minimization of the surface potential function in a specified domain space, X_i, over time is defined as:

$$\nabla_{x_i}\phi_i = \frac{\partial\phi_i}{\partial t}. \tag{1}$$

Each primitive is bound to a selected subset of input control resources $\gamma_j \in \Gamma_j$ and output control resources $\gamma_k \in \Gamma_k$ relevant to the task. In order to bind input and output control resources to the controller, corresponding sensor transforms s_j and effector transforms e_k are used. The sensor transform maps incoming sensory resources to a specified domain space such that $s_j : \Gamma_j \rightarrow X_l$. To ensure that a task is guaranteed to operate within the region of a corresponding basis we require that the output range of a sensor transform matches the artificial potential function domain's data type. Similarly, the effector transform maps the control law error's result to an appropriate output space, Y_k. The mapping is typically effected using a Jacobian matrix as in (2).

$$e_k(\gamma_l) = \left(\frac{\partial \mathbf{x}_{\gamma_1}}{\partial \mathbf{y}_k}, \frac{\partial \mathbf{x}_{\gamma_2}}{\partial \mathbf{y}_k}, \ldots, \frac{\partial \mathbf{x}_{\gamma_{|S_l|}}}{\partial \mathbf{y}_k}\right)^T, \tag{2}$$

where, $\mathbf{x}_{\gamma_i}$ represents the controller update for a sensor control resource γ_i. Also, $\mathbf{y}_k$ is a corresponding point in the output space and $\gamma_l = \gamma_1, \gamma_2, \ldots, \gamma_{|S_l|}$ is a subset of selected control resources for a given task. In order to match an effector transform with an artificial potential function, the rowspace of $e_k(\gamma_l)$, X_i, must match the potential function's data type.

The closed-loop controller is implemented then, when the error between the incoming sensor information and the reference position is minimized within the discrete artificial potential basin, $\nabla_{x_i}\phi_i(\mathbf{x}_{ref} - s_j(\gamma_j))$, and the gradient result is mapped to the output configuration space through an effector transform, $e_k(\gamma_l)$. Given that the input data is of the same domain type as the artificial potential function, and the effector transform is of the same dimensions as the potential function, the controller's output, $\nabla_{y_k}\phi_i$, is defined as:

$$\nabla_{y_k}\phi_i = e_k(\gamma_l)^T \nabla_{x_i}\phi_i(\mathbf{x}_{ref} - s_j(\gamma_j)). \tag{3}$$

For convenience, the above expression is expressed in simplified notation as $\phi_i\,|_{e_k(\gamma_l)}^{s_j(\gamma_j)}(\mathbf{x}_{ref})$. If the controller has zero reference, then it can be omitted: $\phi_i\,|_{e_k(\Gamma_l)}^{s_j(\Gamma_j)}$.

To concurrently optimize multiple goals, secondary control updates are projected onto the nullspace of primary

control updates. This relationship is expressed for a compound controller π as having the secondary controller ϕ_2 be *subject-to* the primary controller ϕ_1, and is expressed as:

$$\nabla_y(\phi_2 \triangleleft \phi_1) = \nabla_y\phi_1 + \mathcal{N}(\nabla_y\phi_1^T)\nabla_y\phi_2, \tag{4}$$

where,

$$\mathcal{N}(\nabla_y\phi_1^T) \equiv I - (\nabla_y\phi_1^T)^+(\nabla_y\phi_1^T), \tag{5}$$

and, I is the identity matrix, y is an n-dimensional space, and the nullspace of $\nabla_y\phi_1^T$ is a $(n-1)$ dimensional space orthogonal to the direction of steepest descent (Platt 2006). For convenience, (4) is written as $\pi_k : \phi_1 \triangleleft \phi_2$.

3 Devising a control basis for cooperative assembly tasks

According to Inuoe (1981), insertion assemblies require that a peg approximates a female fixture with a cautious motion until an optimal location for insertion is achieved; at which point, a compliant insertion takes place to correct misalignments. For this reason, two fundamental compound controllers were designed to drive an insertion: a *Guarded Move Controller* π_{GM} and a *Compliant Insertion Controller* π_{CI}. The former was designed to reactively and autonomously displace a truss to an optimum location for insertion. Upon converging to an optimum locale, a designed control policy transitions to the latter controller to drive the insertion task. These components were used for both the HP3JC robot and ISAC, although for ISAC, the controllers combined the effects of two arms as if they were a virtual clamp. The modified controllers are referred to as the *Virtual Guarded Move Controller* π_{VGM} and the *Virtual Compliant Insertion Controller* π_{VCI}. ISAC also used a third controller designed to counterbalances the forces exerted by an incoming truss and is referred to as the *Virtual Counterbalance Controller* π_{VCB}.

In effect, for the "push-hold" scheme, a control policy is effected such that for the driving robot a guarded move approach drives the truss to an optimum local; once reached, a compliant insertion takes place. For the holding robot, a counterbalance controller is run throughout to optimize the entry of the driving truss. For the "push-push" scheme, both robots reach an optimum locale first and then make use of the compliant insertion controller to drive the insertion (for more details see Sect. 4).

These compound controllers are formulated by combining position ϕ_p, force residual ϕ_{fr}, and moment residual ϕ_{mr}, controllers in different order, with different references, and with different sensor and effecter transforms. The primitive controllers will be presented next and after that how they are compounded.

3.1 Control basis primitives

3.1.1 The position primitive

The position controller is based on the Jacobian transpose control method, where at each cycle, joint displacements are updated according to:

$$\Delta\mathbf{q} = J^T K_p \mathbf{e}, \tag{6}$$

where, $\Delta\mathbf{q}$ is a displacement of joint angles, J^T is the manipulator Jacobian, K_p is the position gain, and $\mathbf{e}$ is the error in cartesian positions.

For the position primitive, the sensor transform converts image coordinates to cartesian positions: $s_p(\gamma_{visual\_sys})$, while the effector transform maps the updated cartesian position to the robot's current joint configuration: $e_p(\gamma_{cart})$. The square of the cartesian error is used as the error function: $\phi_p = \frac{1}{2}K_p\mathbf{e}^T\mathbf{e}$, such that the gradient is:

$$\nabla_x\phi_p = K_p\mathbf{e}. \tag{7}$$

The basis controller can thus be defined as:

$$\nabla_q\phi_p = e_p(\gamma_{cart})^T\nabla_{x_p}\phi_p(\mathbf{x}_{ref} - s_p(\gamma_{visual\_sys})), \tag{8}$$

or, more succinctly as

$$\phi_p\,|_{e_p(\gamma_{cart})}^{s_p(\gamma_{visual\_sys})}(\mathbf{x}_{ref}). \tag{9}$$

3.1.2 The force and moment primitives

Force and moment controllers update joint angle configurations so as to apply desired forces or moments. The force controller updates the first three joints configurations of both robots 6 DoF, while the moment controller updates the last three joint configurations. The joint angle updates for both control laws are defined as:

$$\Delta\mathbf{q}_{1-3} = K_j^{-1}J^T K_f(\mathbf{f}_{ref} - \mathbf{f})^T(\mathbf{f}_{ref} - \mathbf{f}), \tag{10}$$

$$\Delta\mathbf{q}_{4-6} = K_j^{-1}J^T K_m(\mathbf{m}_{ref} - \mathbf{m})^T(\mathbf{m}_{ref} - \mathbf{m}), \tag{11}$$

where, $(\mathbf{f}_{ref} - \mathbf{f})$ and $(\mathbf{m}_{ref} - \mathbf{m})$ are the force and moment errors respectively; and, K_f and K_m are force and moment gains that multiplied by the Jacobian transpose J^T generate torque updates for the appropriate joint configurations. The inverse of K_j is then used to computed the corresponding joint angle updates for each cycle of the force or moment controller.

The force and moment residual controllers have sensor transforms $s_f(\gamma_{force})$ and $s_m(\gamma_{moment})$ that return the F/T sensor data respectively. The artificial potential functions for the force and moment residual functions are proportional to the square of their errors:

$$\begin{aligned}\phi_{fr} &= \frac{1}{2}K_f(\mathbf{f}_{ref} - \mathbf{f})^T(\mathbf{f}_{ref} - \mathbf{f}),\\ \phi_{mr} &= \frac{1}{2}K_m(\mathbf{m}_{ref} - \mathbf{m})^T(\mathbf{m}_{ref} - \mathbf{m}),\end{aligned} \tag{12}$$

and they are differentiated with respect to their joint angle configurations to displace the trusses and minimize residuals:

$$\nabla_q \phi_{fr} = -K_f(\mathbf{f}_{ref} - \mathbf{f}),$$
$$\nabla \phi_{mr} = -K_m(\mathbf{m}_{ref} - \mathbf{m}). \tag{13}$$

The controllers also have effector transforms $e_{fr}(\gamma_{torque})$ and $e_{mr}(\gamma_{torque})$ that converts torque updates into joint angle updates by multiplying the inverse position gains and Jacobian transpose

$$e_{fr}(\gamma_{torque_k}) = K_j^{-1}(J_{\gamma_1}, \ldots, J_{\gamma_{|\gamma_k|}})^T, \tag{14}$$

$$e_{mr}(\gamma_{torque_l}) = K_j^{-1}(J_{\gamma_1}, \ldots, J_{\gamma_{|\gamma_l|}})^T, \tag{15}$$

to produce the following primitive controllers:

$$\phi_{mr}\,|^{s_{mr}(\gamma_{moment})}_{e_{mr}(\gamma_{torque})}:$$
$$\nabla_q \phi_{mr} = e_{mr}(\gamma_{torque})^T \nabla_m \phi_{mr}(\mathbf{m}_{ref} - s_{mr}(\gamma_{moment})), \tag{16}$$

$$\phi_{fr}\,|^{s_{fr}(\gamma_{force})}_{e_{fr}(\gamma_{torque})}:$$
$$\nabla_q \phi_{fr} = e_{fr}(\gamma_{torque})^T \nabla_f \phi_{fr}(\mathbf{f}_{ref} - s_{fr}(\gamma_{force})). \tag{17}$$

3.2 Compound controllers

3.2.1 *Guarded move controller*

The guarded move controller π_{GM} uses ϕ_p as the dominant controller and ϕ_{mr} as the subsidiary controller. The hierarchy of π_{GM} was decided empirically by prioritizing the need for an optimal insertion location. The position controller updates the robot's joint configuration and displaces the truss to a reference location (x_{ref}). The subordinate moment controller minimizes perturbations if contact is made during the trajectory. The position controller receives a 3D reference cartesian position from a stereo visual system that detects color fiducial marks placed at the fixture's tips (Rojas 2009). The compound controller, π_{GM}, is then implemented as:

$$\pi_{GM} = \phi_{mr}\,|^{s_{mr}(\gamma_{moment})}_{e_{mr}(\gamma_{torque})}\,(\mathbf{m}_{ref}) \lhd \phi_p\,|^{s_p(\gamma_{visual\_sys})}_{e_p(\gamma_{cart})}\,(\mathbf{x}_{ref}). \tag{18}$$

3.3 Compliant insertion controller

The compliant insertion controller π_{CI} is activated after the guarded move controller reaches its reference position and is responsible to drive the insertion. The compound controller is naturally reactive.

As stated earlier, experimental practice suggests that aligning trusses takes precedence over fixing its position during the insertion stage. Such alignment is performed by the minimization of moment errors. For this reason, the moment residual primitive ϕ_{mr} is the dominant controller (with no explicit reference parameter), while a force residual controller ϕ_{fr} acts as the subordinate controller. The subordinate force controller was designed in such a way that the a positive force parameter in the x-direction (fx_{ref}) drives the insertion. Higher force parameters update the joint configuration in larger steps making the insertion in effect faster compared to lower force parameters. The force controller will still minimize perturbations and update the joint configuration of the robot to achieve optimal entry positions.

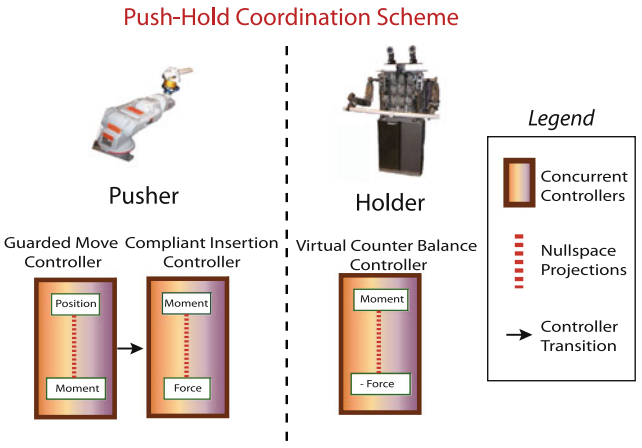


Fig. 2 ISAC and the HP3JC robot use two sequenced compound controllers to implement an assembly using the "push-hold" scheme

In effect, the compliant insertion controller does not use position control to drive the assembly. Instead, it manipulates force reference parameters to create a reactive and adapting controller. The compliant insertion controller is defined in (19). An illustration of the assembly task under a "push-hold" scheme is shown in Fig. 2.

$$\pi_{CI} = \phi_{fr}\,|^{s_{fr}(\gamma_{force})}_{e_{fr}(\gamma_{torque})}\,(\mathbf{f}_{ref}) \lhd \phi_{mr}\,|^{s_{mr}(\gamma_{force})}_{e_{mr}(\gamma_{torque})}. \tag{19}$$

3.4 Virtual clamp and components

To facilitate redundancy problems associated with noise produced from the combined effect of two FT sensors and two arms for ISAC, the concept of a virtual clamp (VC) was used. Both of ISAC's arms were connected through a T-connector to a female truss and acted as a virtual clamp. In effect, the arms, the connector, and the female truss formed a single link. In both, holding and pushing schemes, VC's reduced the control problem from 12 DoF to 6 DoF by averaging joint updates and appropriately updating each of the arms position. The use of the virtual clamp also improved control problems associated with the elasticity and hysteresis of pneumatic muscles (Lefeber 2001). In ISAC, the virtual clamp parameterizes sensor s_i and effector transforms e_k by averaging both sensory stimuli and effector outputs for

given input or output control resources (γ_j, γ_k) in ISAC's right and left arms such that:

$$s_i(\gamma_{vc\_j}) = \frac{1}{2}[s_(\gamma_{j\_left} + \gamma_{j\_right})], \\ e_k(\gamma_{vc\_k}) = \frac{1}{2}[e_(\gamma_{k\_left} + \gamma_{k\_right})]. \tag{20}$$

Virtual clamps were used for the π_{GM} and the π_{CI} controllers as well as for the counterbalance controller. The latter is introduced first followed by the insertion controllers.

3.4.1 Virtual counterbalance controller

The virtual counter balance controller π_{VCB} produces a force guided counterbalance that maintains the static fixture's position in place while updating its orientation and position to facilitate the insertion process and is composed of a dominant moment residual controller and a subordinate force controller with no explicit reference values:

$$\pi_{VCB} = \phi_{fr} \,|^{s_{fr}(\gamma_{force})}_{e_{fr}(\gamma_{torque})} \lhd \phi_{mr} \,|^{s_{mr}(\gamma_{force})}_{e_{mr}(\gamma_{torque})} . \tag{21}$$

The definitions for the virtual guarded move and compliant insertion controllers follow directly from (18) and (19).

$$\pi_{VGM} = \phi_{mr} \,|^{s_{mr}(\gamma_{vc\_moment})}_{e_{mr}(\gamma_{vc\_torque})} \lhd \phi_p \,|^{s_p(\gamma_{vc\_visual\_sys})}_{e_p(\gamma_{vc\_joint})} (\mathbf{x}_{ref}), \tag{22}$$

$$\pi_{VCI} = \phi_{fr} \,|^{s_{fr}(\gamma_{vc\_force})}_{e_{fr}(\gamma_{vc\_pos})} (\mathbf{f}_{ref}) \lhd \phi_{mr} \,|^{s_{mr}(\gamma_{vc\_force})}_{e_{mr}(\gamma_{vc\_pos})} . \tag{23}$$

4 Experiments

Three experiments were conducted under the "push-hold" and "push-push" modes to study the relative performance of the schemes, each robot's proficiency under different roles, and to characterize force signatures of assembly phenomena (jamming, wedging, and stiction) as it pertains to the coupling of a rigid robot and a compliant robot in an autonomous assembly tasks under the already referred-to coordination schemes.

First we describe the testbed, the experimental details, and the performance metrics. Then we describe the experimental procedures. Finally we analyze convergence patterns, the characterization of forces for assembly phenomena with a rigid and compliant robot, relative comparisons between the "push-hold" and the "push-push" schemes, and relative comparisons between the two robots.

4.1 Testbed description

The robotic testbed consisted of Motoman's HP3JC and our in-house built humanoid robot ISAC. The compact HP3JC industrial robot (Motoman 2011) was selected for its high accuracy ability though limited by the inaccessibility to its low-level control. The manipulator used an NX100 Yaskawa controller under an open operating mode in which user-defined control algorithms could be implemented. The controller was housed underneath the robot in a protective case that also served as a mobile base for the robot. Furthermore, a JR3 six-axis F/T and a three fingered Barret Hand were mounted on the wrist. The fingers (and trusses) were covered with adhesive material to reduce friction. ISAC possesses two manipulators each of which is actuated by 12 pneumatic McKibben artificial muscles. The muscles are set-up in agonist-antagonist pairs imitating human muscles (Rojas and Peters II 2005). Each pair of air muscles was driven by 12 air-pressured SMC ITV2050 servo valves. The servo and encoder signals were read through the use of three motion control cards by VitalSys (a more detailed description of the entire humanoid can be found in Rojas 2009). On each of ISAC's end-effectors an ATI six-axis F/T sensor was attached to the wrist as well as a machined aluminum bracket specially designed to grip 1.0 in PVC pipes.

There were two trusses in the experiment: one male and one female, both made from commercial PVC piping. The male truss was constructed by connecting two 0.5 in. pipes through an elbow connector. At the truss' tool-tip an inverted chamfer was used to facilitate its entry into a female counterpart. The HP3JC robot held this truss in all experiments and was placed within ISAC's workspace. The female truss had a T-shape, a 1.0 in. hollow pipe points perpendicularly to the frontal plane of the robot, acting as a female truss. This pipe was connected through a T-connector to two 1.0 in. pipes that were held rigidly by ISAC's machined brackets at the end-effectors. The radial tolerance for the task was 0.25 in. Both trusses had fiducial marks on them to be recognized by a color-based visual system. Figure 3, shows three different perspectives of the experimental setup. In the central image, there is an example of the HP3JC robot before ISAC holding the male truss. The lower left image shows how the Barret hand grips the male truss and approaches the female truss for insertion. The top right image shows how the male truss enters the female truss.

An in-house built decentralized multi-agent architecture known as the Intelligent Machine Architecture (IMA) 2.5 was used to encode all levels of robotic control for both robots including: the visual system, force and moment sensory inputs, and all of the control basis primitives and compound controllers (Olivares 2003). The mobile agent architecture used an event-driven finite-state-machine in conjunction with the control policy to adequately deploy and automate all robotic behavior.

4.2 Experimental details

For all experiments, an insertion was considered complete when the fiducial mark in the male truss was entirely covered (a depth of 2 in). Each experiment began with the

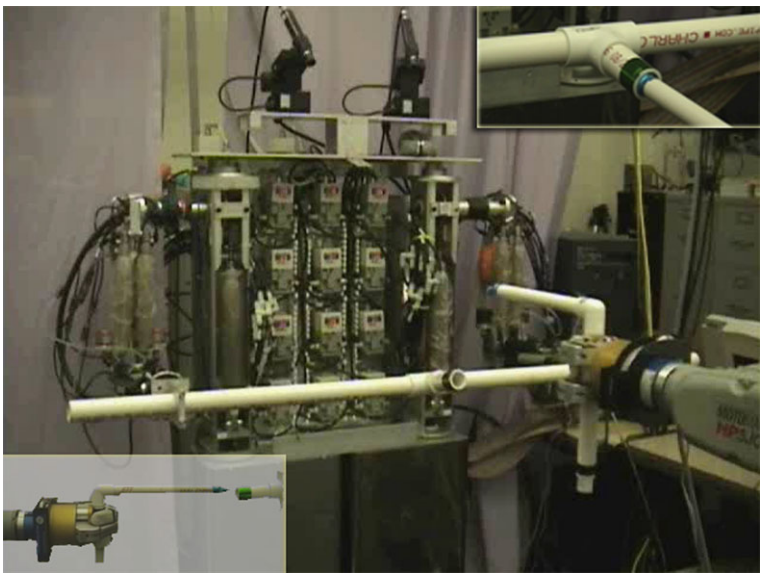


Fig. 3 This three part image illustrates the experimental set-up. The industrial robot is placed before the humanoid robot while the former holds a male truss and the latter a female truss. *The lower left image* is an example of how the trusses look before insertion. *The top right image* is an example of how trusses look during an insertion

HP3JC robot at approximately the same position with a deviation of ±2 in in radius. Color segmentation, as in Rojas and Peters II (2005), was used with empirically set parameters for a low-pass filter and morphological operations. Image processing was used by ISAC to compute a single cartesian coordinate for each of the fiducial marks on the tips of the trusses. Note that visual servoing did not take place in this experiment. Once the initial position reference was provided to the guarded move controller, the visual system did not continue to provide visual feedback. The visual system returned correct reference positions in 25 of 31 trials across three experiments. Each robot's coordinate system was selected by having the z-axis point vertically upwards, the y-axis point perpendicular to both the frontal plane and the Z axis, and the x-axis determined by the right hand rule.

4.3 Performance metrics

Three metrics were used to measure the assembly tasks' performance: (a) time-to-completion, (b) the sum of the absolute values of the maximum moment residuals registered in the x-, y-, and z-directions (referred to hereafter as "moment errors"), and (c) the compliant insertion controller force reference parameter. The first metric was considered to be the time from which the compliant insertion controller starts until the insertion is completed. For the "push-hold" scheme the time was controlled by the driving robot, while in the "push-push" scheme time was controlled by the industrial robot. The second metric took the largest moment errors present at any point in time in the x-, y-, and z-directions and added their absolute value to obtain a single moment error measurement. The third metric differentiated faster and slower insertions driven by the industrial robot (the compliant robot only used one force reference value) to test the system's response under different assembly speeds. Faster insertions used a force reference of $F_x = 178$ N and slower insertions used a value of $F_x = 89$ N. Insertions were assumed complete when the male fiducial mark was covered.

4.4 Experiment 1: HP3JC-push, ISAC-hold

In the first experiment, the industrial robot drove a male truss into a female fixture held by ISAC. The HP3JC robot used the π_{GM} and the π_{CI} controllers to actively perform the insertion, while ISAC used the π_{VCB} controller to counteract, but optimize entry forces exerted by the industrial robot. Fifteen trials were attempted. Three failed due to slight inaccuracies in the cartesian coordinates provided by the visual system. Eight trials were run with the faster force reference parameter $F_x = 178$ N, and four were run with the slower force reference $F_x = 89$ N.

4.5 Experiment 2: ISAC-push and HP3JC-hold

The second experiment reversed the push-hold role play and assigned ISAC as the active robot in the insertion task. This experiment, as opposed to the previous one, began by having the male truss held by the HP3JC at a ready-position in front of the female fixture held by ISAC. The humanoid's virtual-contact compliant insertion controller, π_{VCI} was responsible for driving the insertion. In this experiment, a force reference parameter of $F_x = -2.22$ N was used by the subordinate controller of the virtual compliant insertion controller for ISAC. This experiment carried out seven trials all of which were successful.

4.6 Experiment 3: push-push scheme

The third and last experiment tested the push-push coordination scheme. The HP3JC robot used the π_{GM} and π_{CI}

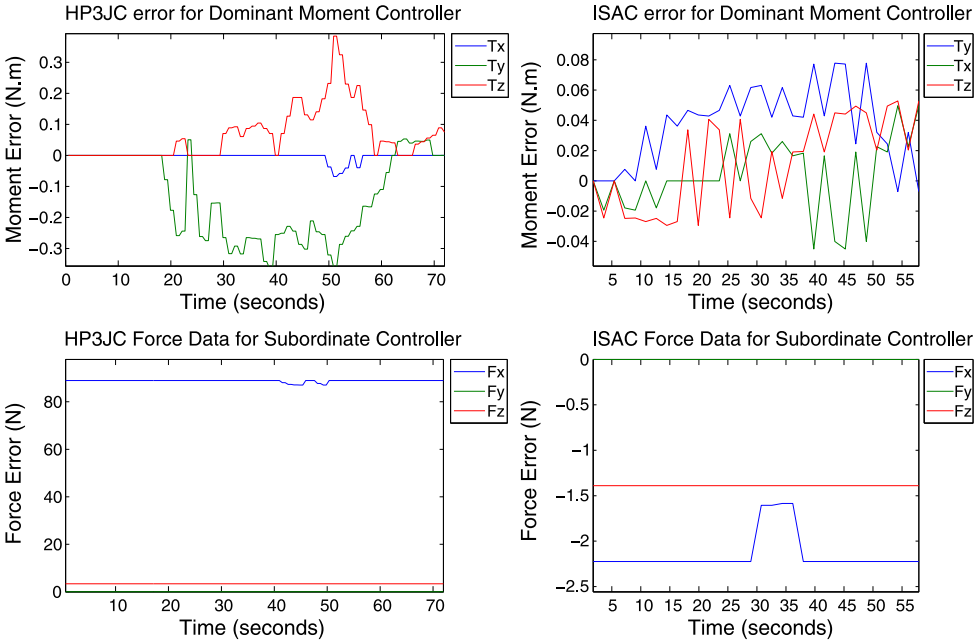


Fig. 4 Trial in Experiment 3 depicting moment residual minimization and force reference parameters serves as a point of reference for other trials

controllers for the HP3JC robot. Nine trial were attempted. Three failed due to inaccuracies in the visual system. Three trials used a force reference of $F_x = 89$ N and the other trials used $F_x = 178$ N. ISAC, on the other hand, used π_{VGM} and π_{VCI} with a reference force of $F_x = -2.22$ N for all trials. The visual system triggered the HP3JC's motion by providing a reference position to the guarded move controller. ISAC remained at the home position using π_{VGM} and transitioned to π_{VCI} upon initial contact.

4.7 Results

4.7.1 Convergence

This section presents three representative moment and force residual plots. The purpose is to demonstrate the effectiveness of the controllers in minimizing the residual of the error in different case scenarios. The first trial corresponds to a "push-push" scheme in Experiment 3. This trial exhibited little stiction and no jamming and served as a baseline for comparison with other trials. The results are shown in a pair of subplots in Fig. 4. Note that the top plots correspond to moment primitives which minimize moment residuals—due to a reference parameter with value of 0. The bottom plots correspond to the force primitives which drive the insertion. In this case, the controller seeks to maintain reference parameter values throughout the task. The second trial corresponded to Experiment 3 which exhibited combined stiction and jamming phenomena, with results shown in Fig. 5. The third trial corresponded to Experiment 1 which exhibited jamming and whose results are shown in Fig. 6.

For the HP3JC robot in Fig. 4(a) the male truss entered the female truss from an offset position both horizontally and vertically giving rise to moment residuals in the Ty and Tz directions. Generally as the male truss begins to enter the female truss it will make contact at one point in the truss. The controller reacts and rejects the disturbance and moves the truss in the opposite direction. The truss then hits the female truss with its tip in an opposite side. The controller again reacts to remove this disturbance, all the while the insertion continues. Eventually there may be two contact points between the male truss and the female truss. In this intermediate stage the controller works to *align* both trusses. At this stage the controller error will tend to oscillate. Then, as the trusses alignment improves, the moment residual constantly decreases. Notice that the controller never fully converges to 0 error. The result was expected. The compliant insertion controller is reactive in nature and rejects disturbances as they appear. Given that the tolerance in our experimental set-up is of 0.25 in, trusses never align perfectly, giving rise to a zig-zag motion of the male truss as it entered the female truss. Note that our termination criteria for the assembly limited deeper insertions and the extent to which the trusses could align. The insertion was nonetheless successful. Better convergence's can be expected for tighter radial tolerances and deeper insertions. With respect to the force data of the HP3JC, a slow reference value was used to drive the insertion with negligible disturbance. This reflects a smooth insertion by the male truss.

The duration of this and other experiments may seem significant. The primary reason for such durations was the inability to access the industrial robot's low-level control loop. Such impediment forced us to operate through the industrial robot's API which prevents pre-emptive motion and significantly delayed the overall response of the system.

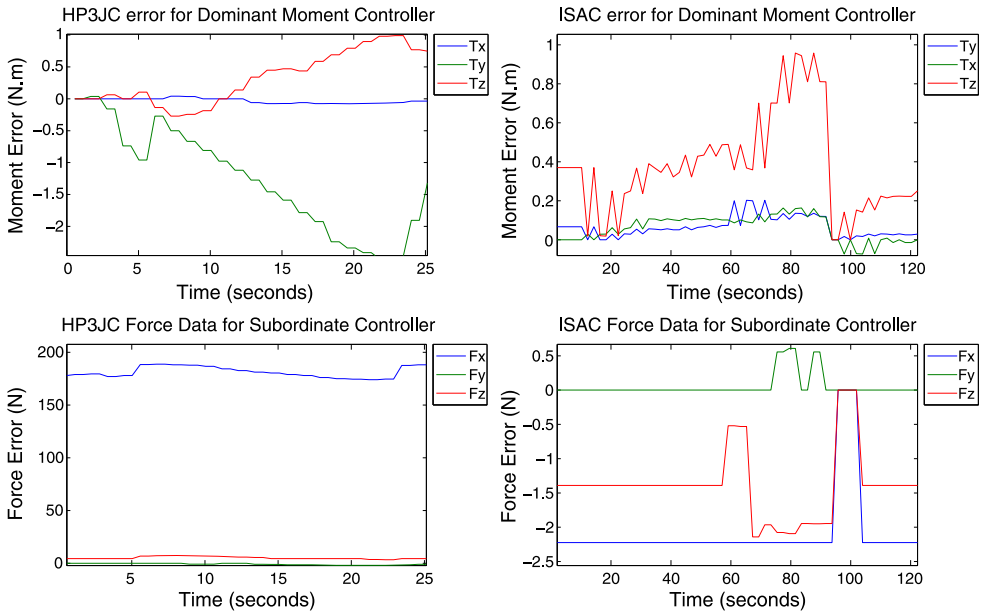


Fig. 5 Trial in Experiment 3 depicting moment residual minimization and force reference parameters in the presence of jamming

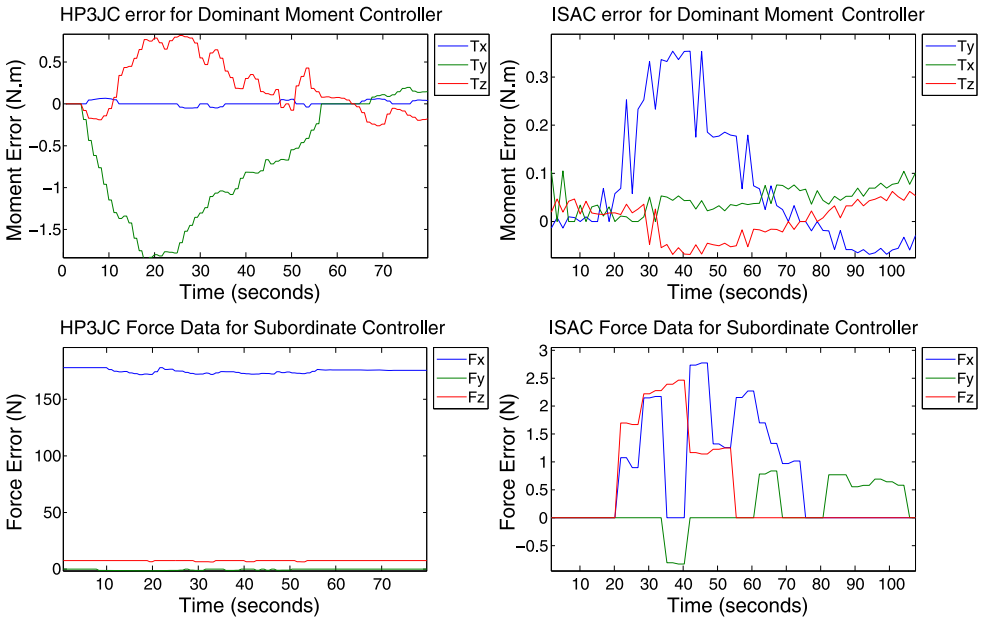


Fig. 6 Trial in Experiment 1 depicting moment residual minimization and force reference parameters in the presence of jamming and stiction

With respect to ISAC, the general pattern of oscillations in the moment residuals is seen here as well. With respect to the force data, ISAC used a force reference parameter $F_x = -2.22$ N. The female truss experienced little stiction (detailed analysis of stiction is presented in Sect. 4.7.2) which led to a halt during its motion. The controller rejected the disturbance and converged to its reference parameter.

The second case corresponded to another trial in Experiment 3. This trial exhibited jamming (a detailed analysis of jamming is presented in Sect. 4.7.2). The moment residual controller struggled to minimize the contact forces generated when the male truss jammed one of the edge walls of the female truss at the entry point. It took the controller about 24 seconds to overcome residuals that grew to more than 89 N-in in the downwards direction. At that point, the moment primitive constantly reduced the disturbance. The forward motion of the male truss was mostly unaltered. For ISAC, jamming phenomena was seen between time markers 80–100 s. Both the moment and force primitives minimized disturbances after a large increase in residual errors. The last representative trial corresponds to a "push-hold" scheme in Experiment 1. This trial exhibited some jamming and stiction. The controllers show good convergence for the HP3JC robot on both the moment and force residuals. For ISAC the moment residual controller minimized the disturbance caused by jamming seen between time ticks 20–60 s, and the

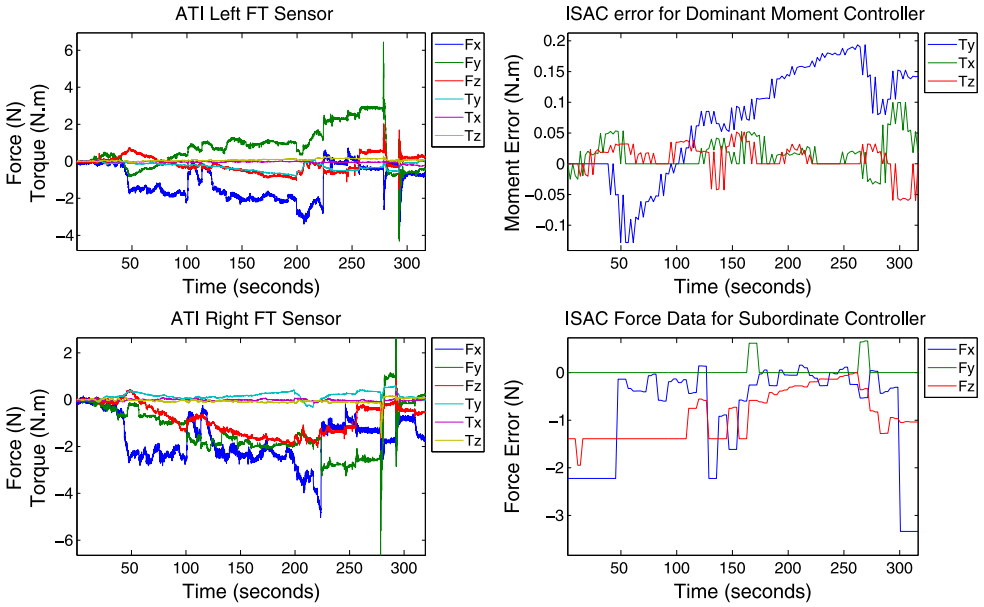


Fig. 7 Trial in Experiment 2. Force and moment plots *on the left*. Moment and force residual plots *on the right* depicting stiction phenomena from the pneumatic humanoid

force residual controller successfully eliminated reappearing and constant disturbances caused by stiction.

4.7.2 *Characterization of forces for assembly phenomena*

This section deals with the characterization of three phenomena: (a) stiction, (b) jamming, and (c) wedging. Stiction phenomena is associated with the actuation of elastic muscles (Lefeber 2002). Stiction as its name implies, refers to a tendency of the pneumatic muscle to stick in a position. In Experiment 2, where ISAC is the pusher, significant stiction was present. Figure 7 correlates the force-moment signatures sensed by the ATI FT sensors on both wrists with the moment and force residuals. In force data plot for the force controller, stiction is present when the forward motion (force parameter $F_x = -2.22$ N) was completely canceled. This occurred twice between 50–75 s and between 200–250 s. Furthermore, it is seen that after these two periods there is a rapid change in force residuals. Near time 135 s, the force changed rapidly from 0 to -2.22 N. This change represents a complete halt of motion followed by a quick forward boost. This stiction point is not readily seen in the FT sensory data but at the time near 300 s stiction points in both arms work in concert to show a swift change in force values from $F_x = -0.44$ N to $F_x = -3.56$ N, while the moment residual plot also showed evidence of quick changes in all three directions. These results match those of Fig. 6(b) where quick changes in force residuals are seen. We noticed that stiction was predominant in trials where the rigid manipulator did not push during the insertion.

Jamming phenomena, as can be seen from Figs. 5 and 6, increased the moment residual in proportion to the force with which the trusses were driven. This phenomena behaved similarly independently of whether a male or female truss was used, or whether a "push-hold" or a "push-push" scheme was used. Jamming, if it occurred, occurred at the beginning of the insertion in which the male truss contacted the edge of circular entry wall of the female truss and for some time pressed against it until the controllers overcame the high contact forces and adjusted the configuration of the system.

Wedging occurs when a peg makes a two-point contact with a female truss and the direction of motion prevents the parts from aligning. One of those contacts, exerts a force at the tip of the male truss inside the female's fixture hull in one direction; the other contact exerts forces at the interface of the edge of the female fixture and the body of the male truss in the opposite direction. In Experiment 1, during a fast insertion, wedging took place yielding the results shown in Fig. 8(a), the male truss shows force signatures that evidence a smooth insertion, but in Fig. 8(b), the moment residual controller for ISAC was tricked into thinking there was no need for realignment. While the male truss was inside the female truss, the controller never converged and the wedge was not resolved. The force residual for the compliant robot presented an intricate pattern of intermittent backwards (positive F_x values) and downwards motions (positive F_z values). Wedging, in general, is more likely in "push-hold" schemes and is characterized by a backward motion in the holding robot.

4.7.3 *Relative comparisons between push-hold and push-push schemes*

This section compares averaged time-to-completion and moment errors from all successful trials in Experiments 1–3 as presented in Sects. 4.4–4.6. These parameters were

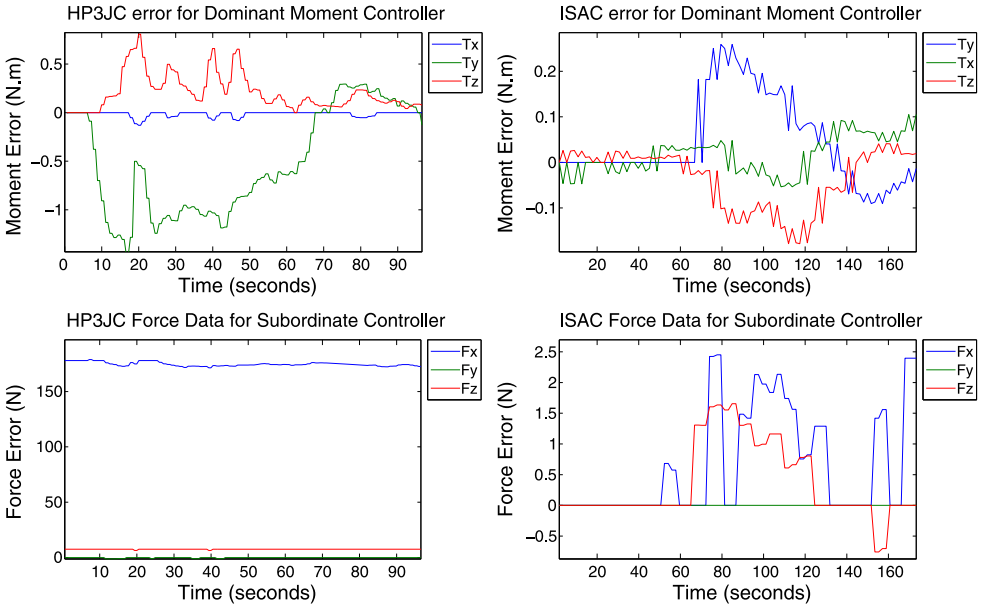


Fig. 8 Robot response in a "push-hold" scheme with wedging. The HP3JC push behavior is shown *on the left*. ISAC's hold behavior is shown *on the right*

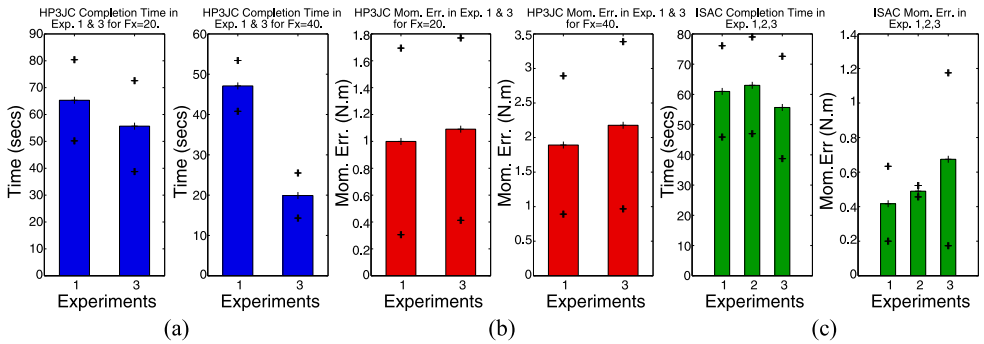


Fig. 9 Error bar comparison between averaged time-to-completion and moment errors for successful trials in Experiments 1–3 under different coordination schemes

grouped according to the force reference value used: $F_x = 89$ N or $F_x = 178$ N for the HP3JC and $F_x = -2.22$ N for ISAC. The reference value set the task's speed and was useful to categorize data into groups belonging to faster insertions or slower insertions. Two trials in Experiment 1 with jamming and wedging registered abnormally high moment errors and time-to-completions and were not considered in this analysis.

For the HP3JC, Experiments 1 and 3 were compared with respect to time and moment errors to study the differences between schemes. For ISAC, Experiments 1, 2, and 3 were compared simultaneously. The results are shown using error bars in Fig. 9.

Figure 9(a) showed indeed that insertions with $F_x = 178$ N were faster, and that the "push-push" scheme was faster. For slow assemblies, Experiment 3 was 9% faster than Experiment 1. For fast assemblies Experiment 3 was 58% faster than Experiment 1.

Figure 9(b) showed that moment errors increased inversely proportional to time-to-completion. Faster insertions had higher error moments. For slow assemblies, Experiment 3 had 9% larger moment errors than Experiment 1. For fast assemblies, experiment 3 had 15% larger moment errors than Experiment 1.

In Fig. 9(c), Experiments 1 and 3 assume timing from the slower insertions for the HP3JC. In Experiment 2, timing comes from ISAC driving the insertion. When ISAC drove the insertion in Experiment 2, it was 3% slower than when the HP3JC drove the slow insertions and 68% slower than when the HP3JC drove the faster insertion. As with the HP3JC, the "push-push" scheme was faster than the "push-hold scheme" driven by ISAC. The moments tell the same story as with the HP3JC, faster assemblies and more collaborative schemes induce greater contact forces on both robots. The controllers react by applying larger corrective steps and completing the insertion faster.

With respect to jamming and wedging phenomena, we saw that in "push-push" schemes, the motion of both trusses diminishes the likelihood of these events from happening. Wedging did not occur in Experiment 3 and when jamming occurred, results showed that contact forces were reduced faster than in the "push-hold scheme".

4.7.4 *Relative comparisons between the rigid robot and the pneumatic robot*

This section studies relative comparisons in the force signatures of the robots during the three experiments as well as their influence on completion times and moment errors.

When comparing the convergence patterns between robots, the trajectories of the HP3JC are smoother than those of ISAC. This data reflects the rigid and compliant natures of the robots. The HP3JC naturally follows its point-to-point motion, while ISAC naturally complies to facilitate the assembly.

The error magnitude also reflects the previous statement in that the HP3JC moment errors are higher than those of ISAC in a given experiment. That is, ISAC's compliance reduces contact forces and experiences lower tension.

The last observation noted that the HP3JC robot moves faster more easily than ISAC. The electrically driven linkages of the HP3JC are more easily controlled than the elastic and antagonist pneumatic muscle-pairs.

5 Discussion

5.1 Coordination schemes relative performance

Our discussion first focuses on the significance of the results in relation to the coordination schemes relative performance. Our results found that the "push-push" coordination scheme was faster. Increased levels of cooperation yielded faster time-to-completion albeit with higher contact forces. The maximum limit for contact forces was not explicitly tested for, but it is set by the robot with the lowest payload limit, in this case the HP3JC which is set at 3 kg (or 29.4 N). Further testing is required in approaching this maximum force limit to understand further the behavior of the controllers and thus the parameters. Additionally, a desirable result of faster completions with higher contact forces was the reduction of undesirable phenomena like jamming and wedging. The finding is encouraging as it affirms that teams of heterogeneous robots can successfully work in assembling parts autonomously and with more flexibility. This is particularly timely as more teams of heterogenous robots with very distinct morphologies and attributes are appearing in research (Sellner et al. 2006; Rehnmark et al. 2005). Specially in the areas of construction or assembly of modular structures (Ueno et al. 2003; Ozaki et al. 2004).

Furthermore, our results suggest that in "push-hold" schemes industrial robots are better pushers and compliant robots are better holders. The smoothness of trajectories for the industrial robot along with its ability to increase speed make electrically-actuated rigid robots a better candidate to push in rigid-compliant robotic pairs. The opposite is also true. The elastic nature of compliant actuators along with their propensity for stiction, make them a better candidate to hold and absorb high contact forces generated by the pushing robot. This finding could be extended to novel human-robot teams in which a human and a robot work assemble parts together (Kruger et al. 2009). NASA, for example, has deployed Robonaut PR2 to space to assist astronauts and with maintenance tasks (Bluethmann et al. 2003). Given that humans like ISAC have elastic muscles, we infer that in human-robot teams where assembly, connection, or insertion tasks take place with a robot like Robonaut, a "push-hold" scheme could be more effective by having the human function as the holder and the electronically actuated robot as the pusher.

5.2 Trends across higher-levels of cooperation

The study presented on this paper performed a detailed analysis on a number of issues concerning the autonomous assembly of parts by a team of heterogeneous robots of different morphologies and attributes using force sensing through a control basis approach. It builds on our previous work which offered a preliminary analysis on the teaming of the rigid industrial robot and the compliant humanoid robot with partial force sensing (Rojas and Peters II 2009) as well as the bootstrapping of the "push-hold" scheme through a decentralized modular architecture (Rojas and Peters II 2011b). Comparing the results of this work with our preliminary work (Rojas and Peters II 2009), where the assembly was performed first with a rigidly held female truss and then with the compliant humanoid with no force sensing, corroborates our current findings that schemes with higher level of cooperation (first the "push-hold" scheme with force sensing and then the "push-push" scheme) have lower time-to-completion and higher likelihoods of completing the insertions.

The authors would like to acknowledge that the number of experiments executed in this work are limited and hence also limit the statistical significance of the results. However, as mentioned previously the results obtained in these three experiments are in agreement with previous work experiments carried out in Rojas (2009). The linearity of the results suggests that with higher-levels of cooperation tasks can be completed faster and it is in this result that the experiments provide value. It is also worth noting, that carrying out experiments with humanoid robots, or teams of robots, still involves a very high level of complexity in its organization, set-up, and execution. In order to integrate the vision control, the image processing, and all the controllers, sensors, and actuators used by the control basis, on the two robotic testbeds in a reactive manner, we had to use a distributed multi-agent architecture which consisted of tens of agents in

parallel to the control basis, and whose control policies are enacted through finite-state machines (see Rojas and Peters II 2011a). Such architecture possess a very high degree of dependability across the distributed agents. While great efforts are spent on fault-tolerance, often times if a vital part of the system fails, the experiment cannot be executed. It is also for this reason, that the implementation and experimental results of this automated coordinated and cooperative assembly across the humanoid robot and the manipulator are valuable.

Additionally, the authors would like to discuss the rate of failure in the experiments as it pertains to the relationship between the controllers and the visual system. As stated in Sect. 4.2, in our control basis approach, we did not do real-time visual servoing. The reason was that the industrial robot had to complete it's assigned instruction before moving to the next target position. The position primitive in the guarded move controllers receives only once a position reference from the visual system. Hence, if the goal point is inaccurate the robots are unable to begin the insertion. Note that there were three failures out of 15 trials in Experiment 1 and three failures out of 9 in Experiment 3. Out of the six failed trials, five of these were due to inaccurate reference positions passed by the visual system and one due to the inability of the controller to resolve a case of wedging in Experiment 1. The failed trials do not demerit the efficacy of the controllers. Every time the controllers (except for the wedging case), were able to perform a successful insertion. The limited efficacy of the visual system is a problem that will be addressed using more robust methods as those used by Sujan and Dubowsky (2005) for a cooperative insertion task.

5.3 Comparison with similar approaches

Our work adds to the growing number of applications that have used the control basis as a developmental robotics control framework including grasping, manipulation, bipedal walking, and flocking (Huber and Grupen 1996; Platt et al. 2006; Antonelli et al. 2010), in an effort to simplify the control problem while researching other problems like autonomy, learning, coordination and cooperation. Approaches as in Caccavale et al. (2001), use modular control to switch tasks across two homogeneous robots. Such frameworks highlight the importance of understanding the relative benefits of robots and tasks to architect better cooperation and coordination in systems.

5.4 Gain considerations

The problem of adaptively adapting the gains of the basis controllers was not studied. The selection of gains was done empirically and behaved similarly to those in Natural Admission Control (Mathewson 1994), whereby there was a trade-off between force and moment controller gains and velocity feedback gains. Lower force and moment gains rendered the system more stable but less responsive. Higher force and moment gains increased the responsiveness of the system but also the amount stiction in the task. It would be interesting to study the responsiveness of the system with adaptive gains. Another alternative is to explore gain optimization by enacting a weighted combination of the compound controllers as in Huntsberger et al. (2004), to find the best set of force parameters and PD gains for the controllers.

Such optimization could lead to another interesting possibility, the reordering of compound controllers in the assembly problem. Nonetheless, the authors believe that the current controller ordering would persist given that the assembly problem is well understood to be segmented into an approach part and an insertion part. On the other hand, we believe the controllers could benefit from additional control primitives. In particular, the compliant insertion controller could be augmented by adding a subsidiary position controller to the force controller such that: $\pi_{CI} = \phi_p \lhd \phi_{fr} \lhd \phi_{mr}$. This last controller, would direct the truss to a reference position while maintaining the reactivity of the controller. By providing a reference position relative to the back wall of the female truss, the augmented controller may better identify the termination of a task and its recovery in the presence of failure.

5.5 Stability considerations

With regards to stability, control basis primitives were originally constructed as provably asymptotically stable navigation functions on their defined space. In configuration space, stable in the sense of Lyapunov and in Cartesian space, stable there as well. Projecting control basis actions into another space through Jacobians can lead to singularities. Such cases are mitigated through nullspace combinations. Pathological type 0 or type 2 critical points are not completely eliminated but can be avoided through empirical design of concurrent combinations of basis controllers or learned combinations. Our work however did not consider learning. The sequencing of controllers was decided based on experience. In Brock et al. (2005), however, learning was used to find the best set of control basis primitives to achieve a number of tasks over time . This study similarly did not consider adaptive control of the gains. In our preliminary work, gains for the underlying potentials were derived empirically. We faced a trade-off between stability and responsiveness. Higher gains yielded more responsiveness but less stability similar to impedance control (Mathewson 1994).

6 Conclusion

This study conducted a comparative analysis between two coordination schemes across a heterogeneous robot team using force sensing. A rigid, point-to-point, industrial robot and a dual-armed pneumatically actuated humanoid assembled parts using a modular control framework under a traditional "push-hold" assembly scheme and a human mimicked "push-push" scheme for hard-to-do assemblies.

The study revealed that as robotic cooperation increased, time-to-completion decreased as did the likelihood of jamming and wedging. The implementation and result affirm the use of heterogenous robot teams to perform "push-push" assemblies autonomously in semi-structured environments.

The study also revealed that in "push-hold" schemes, industrial robots are smoother and faster pushers and compliant robots better attenuate contact forces making them better holders. The notion is important as it can be extended to human-robot teams. It suggests that in insertion tasks humans, due to their inherent compliance, would serve better as holders.

7 Future work

The paper also presented limitations that need to be addressed. In particular, more trials need to be run to increase the statistical significance of this work. Similarly, given than most of the failed trials resulted from an inaccurate cartesian coordinate provided by the visual system, real-time visual servoing needs to be integrated into the system. Finally, it would be interest to study if the results obtained in these experiments extend to other kinds of assembly tasks to study whether or not the results are generalizable.

Acknowledgements We acknowledge R. Grupen and R. Platt for their continued support. This work was supported by NASA grant NNX07AF04G.

References

Antonelli, G., Arrichiello, F., & Chiaverini, S. (2010). Flocking for multi-robot systems via the null-space-based behavioral control. *Swarm Intelligence*, *4*, 37–56.

Bluethmann, W., Ambrose, R., Diftler, M., Askew, S., Huber, E., Goza, M., Rehnmark, F., Lovchik, C., & Magruder, D. (2003). Robonaut: A robot designed to work with humans in space. *Autonomous Robots*, *14*, 179–197.

Brock, O., Fagg, A., Grupen, R., Platt, R., Rosenstein, M., & Sweeney, J. (2005). A framework for learning and control in intelligent humanoid robots. *International Journal of Humanoid Robotics*, *2*(3), 301–336.

Brogardh, T. (2007). Present and future robot control development—an industrial perspective. *Annual Reviews in Control*, *31*(1), 69–79.

Caccavale, F., Natale, C., Siciliano, B., & Villani, L. (2001). Achieving a cooperative behavior in a dual-arm robot system via a modular control structure. *Journal of Robotic Systems*, *18*(12), 691–699.

Doggett, W. (2002). Robotic assembly of truss structures for space systems and future research plans. In *Proc. of the 2002 aerospace conf.* (Vol. 7, pp. 3589–3598).

Huber, M. (2000). *A hybrid architecture for adaptive robot control*. Ph.D. dissertation, U. of Mass.

Huber, M., & Grupen, R. A. (1996). *A hybrid discrete event dynamic systems approach to robot control* (Tech. Rep. 96–43). U. of Mass., Department of Computer Science, 140 Governors Drive, Amherst, MA 01003-9264.

Huntsberger, T., Trebi-Ollennu, A., Aghazarian, H., Schenker, P., Prijanian, P., & Nayar, H. (2004). Distributed control of multi-robot systems engaged in tightly coupled tasks. *Autonomous Robots*, *17*, 79–92.

Inuoe, H. (1981). Force feedback in precise assembly tasks. In *Artificial intelligence: an MIT perspective* (Vol. 2, pp. 219–241).

Kruger, J., Lien, T., & Verl, A. (2009). Cooperation of human and machines in assembly lines. *CIRP Annals Manufacturing Technology*, *58*(2), 628–646.

Lefeber, F. D. D. (2001). The concept and design of pleated pneumatic artificial muscles. *International Journal of Fluid Power*, *2*(3), 41–50.

Lefeber, F. D. D. (2002). Pneumatic artificial muscles: Actuators for robotics and automation. *European Journal of Mechanical and Environmental Engineering*, *47*, 10–21.

Mathewson, B. B. (1994). *Integration of force strategies and natural admittance control*. Ph.D. thesis, Case Western U.

Motoman (2011). http://motoman.com/.

Olivares, R. (2003). *The intelligent machine architecture 2.5: A revised development environment and software architecture*. Master's thesis, Vanderbilt U.

Ozaki, F., Machida, K., & Iwata, T. (2004). Robot control strategy for in-orbit assembly of a micro satellite. *Advanced Robotics*, *18*(24), 199–222.

Platt, R. J. (2006). *Learning and generalizing control-based grasping and manipulations skills*. Ph.D. thesis, U. of Mass. Amherst.

Platt, R., Grupen, R., & Fagg, A. (2006). Improving grasp skills using schema structured learning. In *IEEE intl. conf. on development and learning*.

Pratt, G., & Williamson, M. (1995). Series elastic actuators. *International Conference on Intelligent Robots and Systems*, *1*, 399.

Rehnmark, F., Ambrose, R., Kennedy, B., Diftler, M., Mehling, J., Bridgwater, L., Radford, N., Goza, S., & Culbert, C. (2005). Innovative robot archetypes for in-space construction and maintenance. *AIP Conference Proceedings*, *746*(1), 1070–1077.

Rojas, J. (2009). *Autonomous cooperative assembly by force feedback using a control basis approach*. Ph.D. thesis, Vanderbilt U.

Rojas, J., & Peters, R. A. II (2005). Sensory integration with articulated motion on a humanoid robot. *Applied Bionics and Biomechanics*, *2*(3–4), 171–178.

Rojas, J., & Peters, R. A. II (2009). Preliminary results in force guided assembly for teams of heterogeneous robots. In *SPIE's defense, security, and sensing*.

Rojas, J., & Peters, R. A. II (2011a). Cooperative and coordinated assembly for heterogeneous robots via distributed mobile agents. In *Workshop. of the IEEE intl. conf. on robots and automation*.

Rojas, J., & Peters, R. A. II (2011b). Automating short-term insertion of parts for heterogeneous robots using a control basis approach. In *Proc. of the IEEE-RAS intl. conf. on humanoid robots*.

Sellner, B., Simmons, R., & Singh, S. (2005). *Proc. from the intl. workshop on multi-robot systems from swarms to int. automata* (Vol. III). Berlin: Springer.

Sellner, B., Heger, F. W., Hiatt, L. M., Simmons, R., & Sing, S. (2006). Coordinated multiagent teams and sliding autonomy for large-scale assembly. *IEEE Special Issue on Multirobot Systems*, *94*(7), 1425–1444.

Shah, J. A., Saleh, J. H., & Hoffman, J. A. (2007). Review and synthesis of considerations in architecting heterogeneous teams of humans and robots for optimal space exploration. *IEEE Transactions on Systems, Man, and Cybernetics*, *37*(5), 779–793.

Simmons, R., Singh, S., Heger, F. W., Hiatt, L. M., Koterba, S. C., Melchior, N., & Sellner, B. P. (2007). Human-robot teams for large-scale assembly. In *Proc. of the NASA science technology conf.*

Sujan, V., & Dubowsky, S. (2005). Visually guided cooperative robot actions based on information quality. *Autonomous Robots*, *19*, 89–110.

Ueno, H., Nishimaki, T., Oda, M., & Inaba, N. (2003). Autonomous cooperative robots for space structure assembly and maintenance. In *Proc. of the 7th intl. symposium on artificial intelligence, robotics and automation in space*.

J. Rojas is a post-doctoral associate at Japan's National Institute of Advanced Science and Technology (AIST) where he is currently researching robotic automation through signal interpretation and probabilistic learning. He received a B.S., M.S., and Ph.D. in Electrical and Computer Engineering from Vanderbilt University in 2002, 2004, and 2009 respectively. From 2009–2011 he served as a visiting scholar at Sun Yat-Sen University in China. His research interests include long-term and short-term automation, cooperation and coordination of robot teams and human-robot interaction. Juan is also very interested in humanitarian applications and endeavors to find solutions to problems through technology and service.

Richard A. Peters II (a.k.a. Alan Peters) is an associate professor of electrical engineering at Vanderbilt University and the Chief Technology Officer of Universal Robotics, Inc. He is a Phi Beta Kappa graduate of Oberlin College (Ohio) where he received an A.B. in Mathematics (May 1979). He received an M.S. (1985) and a Ph.D. (1988) in electrical engineering from the University of Arizona, where he was a fellow of the American Electronics Association. From 1990–1994 he worked with the US Air Force Office of Scientific Research, Arnold Engineering and Development Center, Arnold Air Force Base, TN as a faculty fellow and consultant. In 1997 he was an NSF/JSTA Fellow at the Electro-Technical Laboratory in Tsukuba, Japan. From 2000–2008 he was a summer faculty fellow on the Robonaut Project at NASA Johnson Space Center, Houston, TX. He was on leave from Vanderbilt 2008–2010 to start Universal Robotics, a software company that designs intelligent, reactive controllers for industrial robots. His current research interests include intelligent robotics, image processing, and computer vision.