

RESEARCH ARTICLE

Open Access



# Hardware-in-the-loop simulation of massive-payload manipulation on orbit

Jaesung Yang<sup>1\*</sup> , Atsushi Konno<sup>1</sup>, Satoko Abiko<sup>2</sup> and Masaru Uchiyama<sup>3</sup>

## Abstract

This paper describes a hardware-in-the-loop simulation of massive-payload manipulation on orbit using a master–slave teleoperation system. The main problems in teleoperating a space robot arm from the earth to manipulate a massive payload are communication delay, unexpected excessive force generated between the slave arm and the payload, and geometric/dynamic modeling error. In order to overcome those problems, a teleoperation system using mixed force and motion commands is discussed. The teleoperation system is verified by performing hardware-in-the-loop simulations.

**Keywords:** Teleoperation, Hardware-in-the-loop simulation, Virtual environment, Massive object

## Introduction

Space robotic manipulators such as SRMS (Shuttle Remote Manipulation System) and SSRMS (Space Station Remote Manipulator System) have played a vital role in satellite orbit injection and construction of the ISS (International Space Station). These space robots have been operated by astronauts on-site.

However, if such manipulation can be performed by teleoperating the space robot from the earth, there is no need to send astronauts to space, which is attractive from both safety and financial consideration.

In general, satellites or modules of space structures have huge mass compared to the robot arm. Astronauts must be very careful when operating the robot arm not to generate excessive force between the end effector of the robot arm and the massive payload being manipulated, especially when starting and stopping the movement. The main problems in the manipulation of a massive payload by teleoperating the robot arm from the earth will be communication delay and the unexpected excessive force generated between the space robot and the payload.

It is well known that a communication delay between a master arm and a slave arm greatly reduces the

performance of the teleoperation [1]. In general teleoperation (e.g. [2]) in which the master and the slave arms are directly coupled, an operator must take “move and wait” strategy, in which the operator must wait the delayed information of the slave-arm to confirm at every slight operation if the communication delay is large. Semi-autonomy is a solution for efficient teleoperation under communication delay [3]. Model-based teleoperation via a virtual environment has also been enthusiastically studied [4–6]. However, in model-based teleoperation via a virtual environment, an unexpected excessive force could be generated by a modeling error in the geometric or dynamic properties of the slave arm and the payload.

In order to overcome the problem of modeling error in the teleoperation, the use of mixed force and motion commands has been proposed [6, 7]. However, model-based teleoperation assumes that the master and the slave arm have similar sizes, and therefore this teleoperation system cannot be directly applied to massive-payload manipulation. In [8], a small size master-arm was used to operate the ETS-VII manipulator, which was an actual space robot. However, only the contact force was fed-back to the operator, and hence force scaling was not considered. The power scaling was considered in [2], however the method proposed in [2] has the problem of inefficiency in teleoperation if the communication delay is large, as discussed above. In this work, model-based teleoperation is modified considering the difference in

\*Correspondence: jsyang@scc.ist.hokudai.ac.jp

<sup>1</sup> Graduate School of Information Science and Technology, Hokkaido University, Kita 14 Nishi 9, Kita-ku, Sapporo, Japan

Full list of author information is available at the end of the article

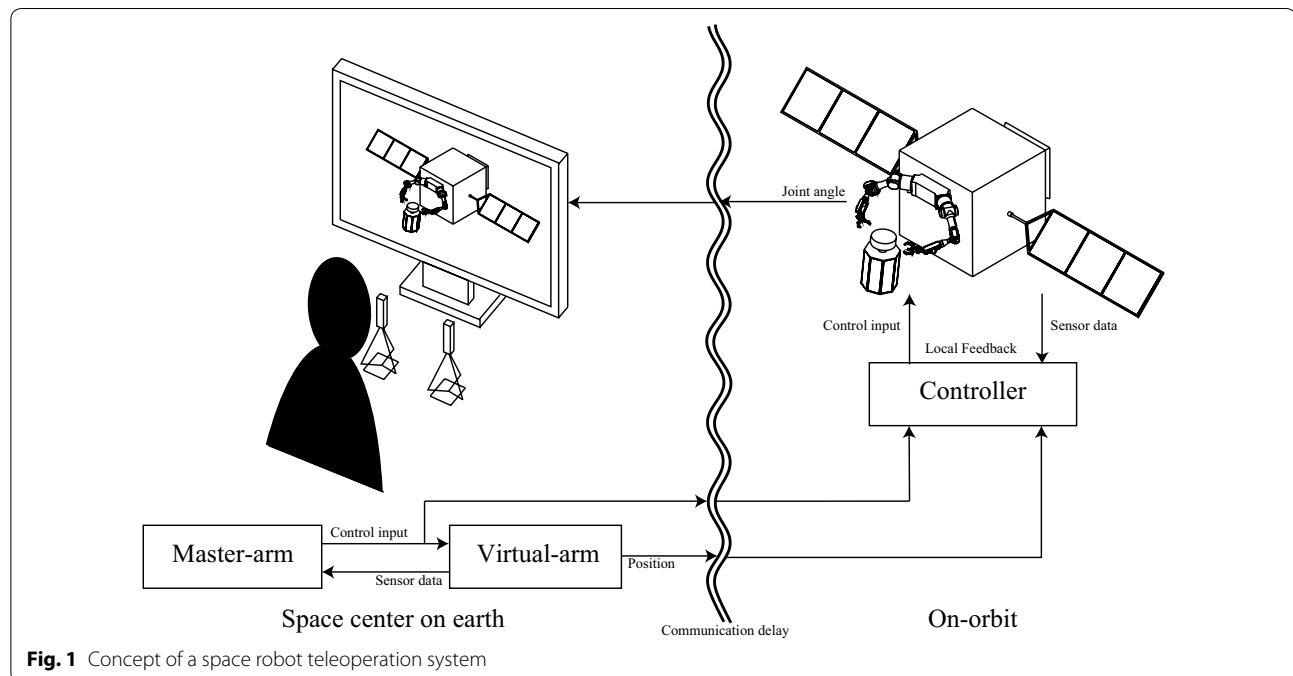
size between the master and the slave arms for application to massive-payload manipulation.

Furthermore, the modified model-based teleoperation is verified using a hardware-in-the-loop simulation (HILS). Various HIL simulators for space robots were surveyed in [9]. HIL simulators for space robots have been developed mainly to simulate the capture of space targets such as satellites [10–12]. However most HIL simulators for space robots have used conventional serial robots to reproduce the movement of the space target. Serial robots do not generally move very quickly, and hence the response delay is relatively large. The large response delay of the motion table (the robot arm used to reproduce the movement of the space target) may cause instability in HILS [13]. Hence, this work uses a parallel-robot-based HIL simulator [14], applying delay time compensation [13] in order to validate the model-based teleoperation by simulating massive-payload manipulation.

The main contributions of this paper are the proposal for a modified version of model-based teleoperation using mixed force and motion commands and the parallel-robot-based HILS of the massive payload manipulation. The proposed method deals with force scaling. In force scaling, mitigation of excessive force at the remote site will be the key issue, because the operator's commands are scaled-up at the remote site, and there is a serious communication delay. The mitigation of generated force at the remote site using the proposed method was verified by performing HIL simulations.

### Hardware-in-the-loop (HIL) space robot simulator

The concept of a space robot teleoperation system is illustrated in Fig. 1. In the space center on earth, an operator manipulates the master arms by looking at the virtual environment and predicting the movement of the space robot on orbit. The control inputs from the master arms and the position data of the virtual arm are transferred to the slave arm on orbit with a communication delay. The slave arms on orbit follow the motion commands generated by the master arms on earth. When the slave arms contact with the payload or the environment, the contact force is locally controlled using force sensor data. However, the slave arms' position data arrive at the space center on earth with a communication delay. Hence, the positions of the slave arms at the moment of teleoperation are unknown on earth. In order to overcome the communication delay problem, simulated motions of the slave arms (virtual arms) are displayed for the human operator. The simulated motion is transferred to the slave arm controller as well as the motion command generated by the master arms (see Fig. 1). The motion of the virtual arms modifies the slave arm's position in order to eliminate the error between the slave arm and virtual arm. The joint angle data of the slave arm are transferred to the virtual environment in order to display the movement of the slave arm, which is superimposed on the movement of the virtual arm. The system can be divided into the hybrid motion simulator, the slave system, and the master system.

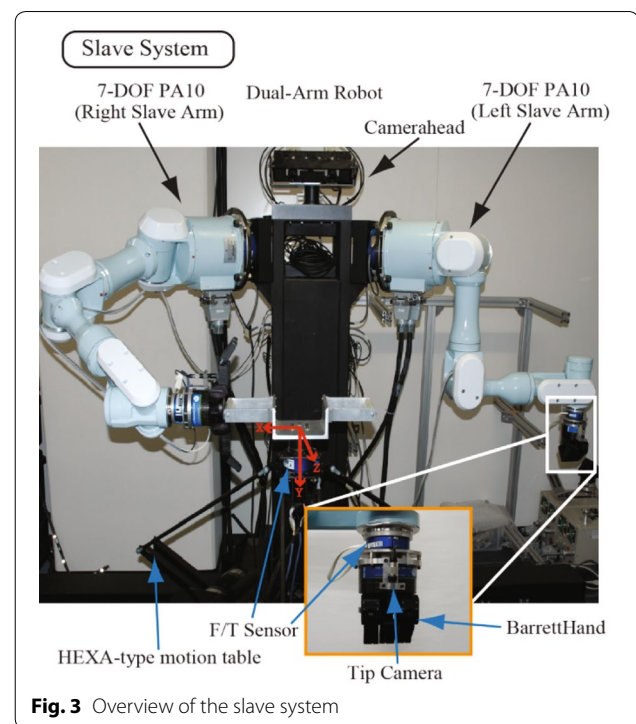


### HIL simulator

The HIL simulator [14] is composed of a 6 axis force sensor for acquiring the external force, an X-Y motion table for reproducing the translational large movement of the payload, and a HEXA-type [15] motion table for reproducing 6-DOF guide movement. The force sensor is mounted at the boundary of the hardware simulation and software simulation. In the HIL simulation, a force and torque acting on the payload are measured by the force/torque (F/T) sensor. The relative position and orientation with respect to the payload of interest are calculated by solving a dynamic equation with the measured force/torque data. Finally, the calculated relative position and orientation are realized on the hardware side by a servo-mechanism in real time (see Fig. 2). The response delay time compensation method [13] is applied to the hybrid simulator.

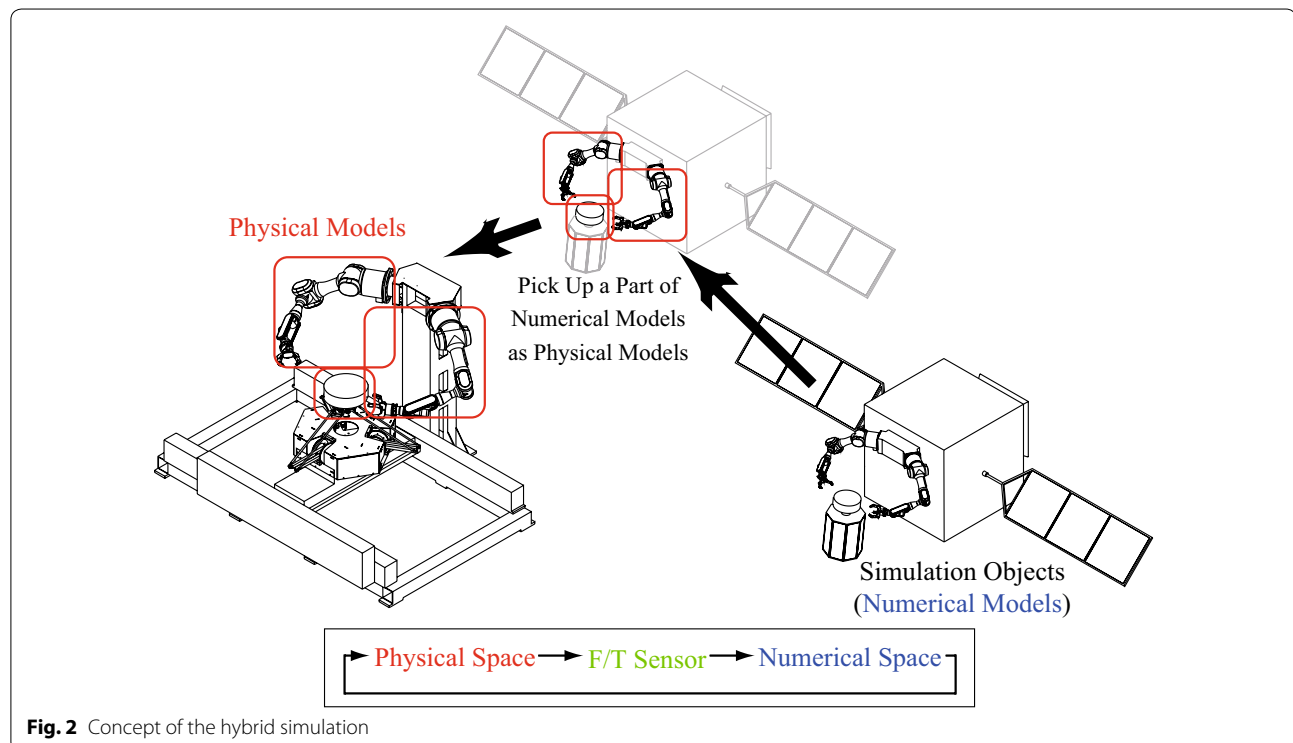
### Slave system

The slave system is composed of two 7-DOF manipulators with a serial link mechanism and a camera head system for monitoring the working environment (see Fig. 3). The camera head can pan and tilt to see the workspace. A wrist camera, a 6-axis force sensor and a BarrettHand™ (Barrett Technology, LLC) for grasping the various payloads are mounted on each end effector. The 6-axis force sensor acquires the force generated at the wrist of the slave arm. The BarrettHand is able to



**Fig. 3** Overview of the slave system

open and close fingers to grip a payload. The end effector camera is used to obtain detailed information with a different field of view angle of the camera head.



**Fig. 2** Concept of the hybrid simulation

### Master system

The master system (see Fig. 4) is composed of two 6-DOF haptic interfaces with a parallel link mechanism [16]. The force sensor used in the 6-DOF haptic interface is NANO 5/4-A (BI Autotec). The 6-DOF haptic interface is used in the joystick mode (the haptic device does not move and does not display forces in the joystick mode) in this work. The virtual environment is displayed on a monitor.

### Development of virtual environment and control law

#### Operation scenario

The assumed task is to convey a massive payload by teleoperating a space robot. A grip mounted on the HEXA-type motion table (Fig. 3) is assumed to be a part of the massive payload. The force applied to the grip is measured by an F/T sensor mounted between the grip and the HEXA-type motion table. The measured force and torque are considered to be the external force applied to the virtual massive payload. The motion of the massive payload is calculated by solving a dynamic equation, and the motion is reproduced by the HEXA-type motion table mounted on the X-Y table.

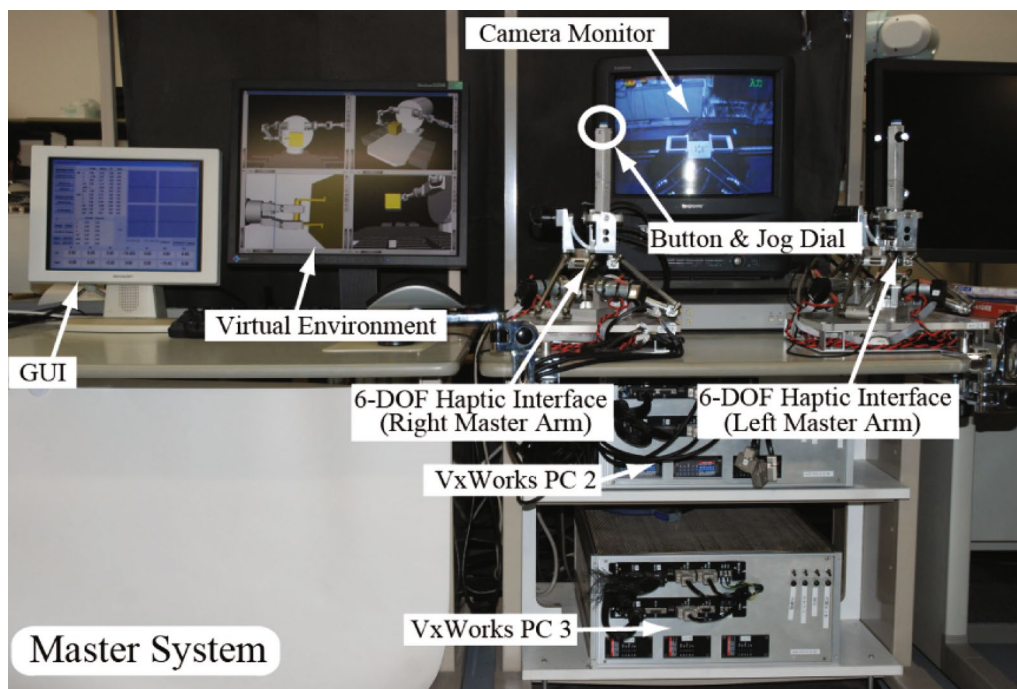
In the operation scenario, it is assumed that a human operator on earth teleoperates a space robot arm (slave arm) to manipulate a massive payload.

### Control law

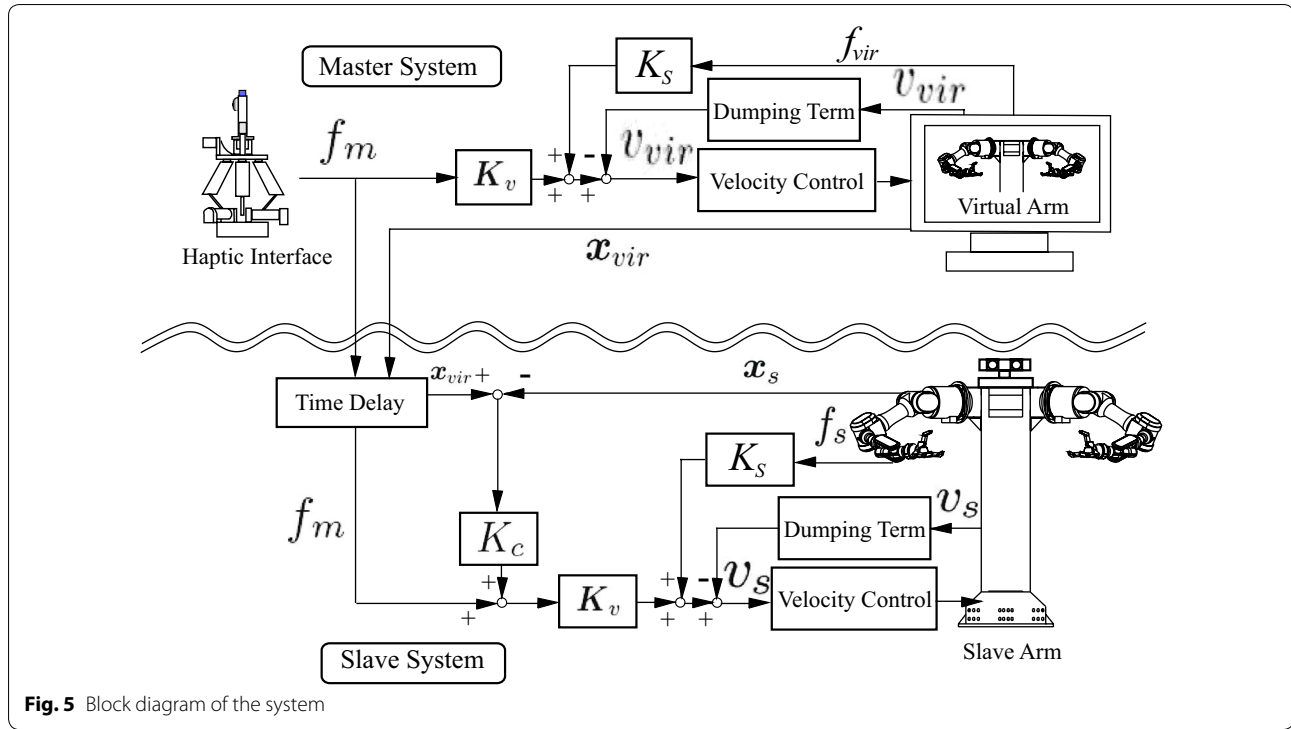
In this work, the haptic device shown in Fig. 4 is used as a master arm in the joystick mode. The force applied by the human operator to the stylus of the master arm is measured by an F/T sensor and is transformed into a velocity command of the slave arm. A block diagram of the teleoperation system is illustrated in Fig. 5.

Let  $\mathbf{x}_i = [\mathbf{r}_i^T \ \boldsymbol{\phi}_i^T]^T$  and  $\mathbf{v}_i = [\dot{\mathbf{r}}_i^T \ \boldsymbol{\omega}_i^T]^T$  ( $i = m, vir, s$ , or  $t$ ) be the six dimensional position and velocity vectors, respectively. Subscripts  $m$ ,  $vir$ ,  $s$ , and  $t$ , respectively, represent the master arm, virtual arm, slave arm, and massive payload.  $\boldsymbol{\phi}_i$  and  $\boldsymbol{\omega}_i$  denote a set of three variables for orientation (e.g., Euler angles) and the angular velocity vector, respectively. The derivative of  $\mathbf{x}_i$  is transformed into  $\mathbf{v}_i$  by the matrix  $\mathbf{B}(\boldsymbol{\phi}_i)$  as  $\mathbf{v}_i = \mathbf{B}(\boldsymbol{\phi}_i)\dot{\mathbf{x}}_i$ .

In the teleoperation with a communication delay, both geometric and dynamic modeling errors are serious problems. If there are very large discrepancies between the real environments at the remote site and their models, it may not be possible to catch/contact a payload with the slave arm via teleoperation, because the operator on earth would not be able to monitor the remote site in real time. Large discrepancies between the dynamic properties of the slave arm/payload and those of the models may cause an unexpected excessive force to be generated between the slave arm and the payload. In order to overcome those problems, model-based teleoperation



**Fig. 4** Overview of the master system



**Fig. 5** Block diagram of the system

using mixed force and motion commands has been proposed [6, 7].

This work uses model-based teleoperation [6, 7], but it is modified to apply to massive payload manipulation. The main modification pertains to the scaling between the master arm and the slave arm. The original model-based teleoperation using mixed force and motion commands [6, 7] assumed that the force and motion in both arms (master and slave arms) were of the same scale. However, in massive-payload manipulation, the force/motion command given by a human operator should be scaled up so that the large-sized slave arm can manipulate the massive payload. In the following sections, modified model-based teleoperation using mixed force and motion commands is presented.

#### Control law of slave arm

The velocity command to the slave arm,  $v_{s,c}(t)$ , is given as follows:

$$v_{s,r}(t) = K_s G f_m(t), \quad (1)$$

$$\begin{aligned} v_{s,c}(t) = & v_{s,r}(t - t_{delay}) - K_s f_s(t) \\ & + K_c B(x_{vir}(t - t_{delay}) - x_s(t)) \\ & - D(v_s(t - \Delta t) - v_s(t - 2\Delta t)), \end{aligned} \quad (2)$$

where  $f_m$  is the reference force commanded by the human operator using the master arm,  $f_s$  is the actual

force generated at the slave arm,  $G$  is a diagonal scaling matrix between the force commanded by the master arm and the force the slave arm is expected to generate. In order for the slave arm to manipulate the massive payload, the force command  $f_m$  given by the human operator must be scaled up for the slave arm. The matrix  $G$  scales up the reference force. Substituting (1) into (2), the following equation is obtained:

$$\begin{aligned} v_{s,c}(t) = & K_s (G f_m(t - t_{delay}) - f_s(t)) \\ & + K_c B(x_{vir}(t - t_{delay}) - x_s(t)) \\ & - D(v_s(t - \Delta t) - v_s(t - 2\Delta t)), \end{aligned} \quad (3)$$

$v_{s,r}$  is the reference velocity given by (1).  $x_{vir}$  is the position vector of the slave arm. They are transferred from the earth and arrive at the space robot  $t_{delay}$  later. Hence,  $f_m$  and  $x_{vir}$  used in (3) are  $t_{delay}$  ahead of the local (space) time.  $x_s$  is the position vector of the slave arm,  $K_c$  is a gain matrix for the restraint force,  $f_s$  is a force vector measured by the F/T sensor of the slave arm, and  $K_s$  is a diagonal force feedback gain matrix.  $D$  is a diagonal damping gain matrix.  $\Delta t$  is the sampling period of the controller. The original model-based teleoperation using mixed force and motion commands [6, 7] was not always stable in HILS, and hence the damping term (the last term of the right hand side of 2) is added in this work.



### Virtual arm

Model-based virtual robots between the real robot and the master arm have been widely used to overcome communication delay (e.g., [6]), or unexpected events (e.g., [17, 18]).

In this work, the virtual arm is used to compensate for the modeling error in dynamic properties as well as to overcome the communication delay. The difference in the positions between the virtual slave arm and the real slave arm is feedback to the real slave arm control, as shown in the second term of the right-hand side of (2). The term helps reduce the positioning error caused by the modeling error in the dynamic properties of the slave arm. The reference velocity of the slave arm is also given to the virtual arm as follows:

$$\mathbf{v}_{vir,r}(t) = \mathbf{v}_{s,r}(t) = \mathbf{K}_s \mathbf{G} \mathbf{f}_m. \quad (4)$$

Similar to the velocity command to the slave arm (2), the velocity command to the virtual arm is given as follows

$$\mathbf{v}_{vir,c}(t) = \mathbf{v}_{vir,r}(t) - \mathbf{K}_f \mathbf{f}_{vir}(t) - \mathbf{D}(\mathbf{v}_{vir}(t - \Delta t) - \mathbf{v}_{vir}(t - 2\Delta t)), \quad (5)$$

$$\mathbf{x}_{vir}(t) = \mathbf{x}_{vir}(t - \Delta t) + \mathbf{B}^{-1} \mathbf{v}_{vir}(t) \Delta t, \quad (6)$$

where  $\mathbf{v}_{vir}$  is the velocity of the virtual arm. The matrix  $\mathbf{B}^{-1}$  transforms  $\mathbf{v}_{vir}$  into  $\dot{\mathbf{x}}_{vir}$ . The virtual arm control law is almost the same as the control law of the slave arm to predict the movement of the slave arm and present it in the virtual environment. When the slave arm employs high-ratio reduction gears at the joints,  $\mathbf{v}_s$  can be approximated by  $\mathbf{v}_{s,c}$  (see Appendix A). Therefore, the velocity of the virtual arm,  $\mathbf{v}_{vir}$ , is approximated by  $\mathbf{v}_{vir,c}$ .

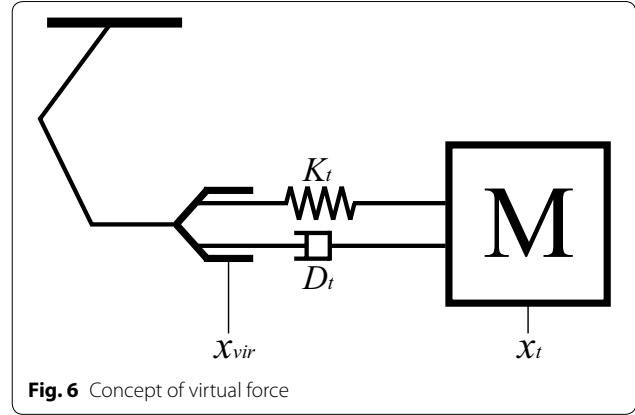
In the virtual environment, the contact force between the massive payload and the end effector of the slave arm is simulated by a spring-damper system, as illustrated in Fig. 6.  $\mathbf{f}_{vir}$  is simulated as follows:

$$\mathbf{f}_{vir}(t) = -\mathbf{K}_t \mathbf{B}(\mathbf{x}_{vir}(t - \Delta t) - \mathbf{x}_t(t - \Delta t)) - \mathbf{D}_t(\mathbf{v}_{vir}(t - \Delta t) - \mathbf{v}_t(t - \Delta t)), \quad (7)$$

where  $\mathbf{K}_t$  is a spring constant matrix,  $\mathbf{D}_t$  is a viscosity coefficient matrix,  $\mathbf{x}_t$  is a position vector of the virtual massive payload, and  $\mathbf{v}_t$  is a velocity vector of the payload. The acceleration of the massive payload is expressed by

$$\dot{\mathbf{v}}_t(t) = \mathbf{M}^{-1} \mathbf{f}_{vir}(t), \quad \mathbf{M} = \begin{bmatrix} m\mathbf{E} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \quad (8)$$

where  $m$  is the mass,  $\mathbf{E} \in \mathbb{R}^{3 \times 3}$  is an identity matrix, and  $\mathbf{I} \in \mathbb{R}^{3 \times 3}$  is the inertia tensor of the massive payload. The massive payload will be a satellite or a module of a space structure, and hence  $m$  and  $\mathbf{I}$  are known (they can be obtained from the design parameter). However,



the inertia matrix  $\mathbf{M}$  may include a modeling error. The position of the payload is estimated by (8–10), hence the modeling error in  $\mathbf{M}$  in (8) causes an estimation error. The position error caused by the modeling error is compensated for by the coupling term in the slave arm controller (the third term of the right hand side of 2).

The velocity and position of the massive payload are calculated by

$$\mathbf{v}_t(t) = \mathbf{v}_t(t - \Delta t) + \dot{\mathbf{v}}_t(t) \Delta t, \quad (9)$$

$$\mathbf{x}_t(t) = \mathbf{x}_t(t - \Delta t) + \mathbf{B}^{-1} \mathbf{v}_t(t) \Delta t. \quad (10)$$

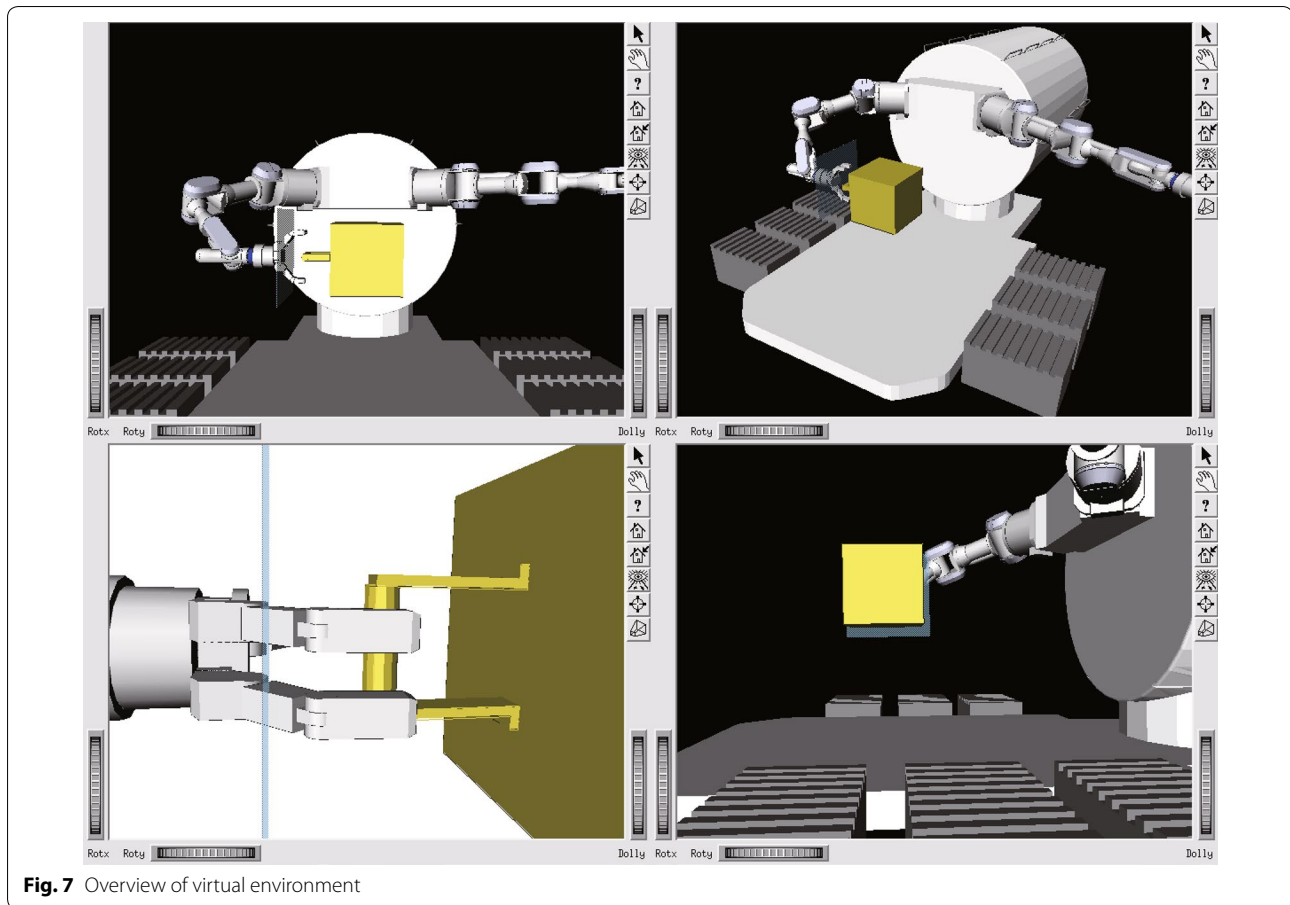
### Virtual environment

As discussed in Section "Virtual arm", the velocity of the virtual arm,  $\mathbf{v}_{vir}$ , was approximated by  $\mathbf{v}_{vir,c}$  assuming high reduction gear ratios at the joints. Only the dynamics of the massive payload was considered in the virtual environment (7–10). The dynamics of the massive payload, (7–10), is used in the computation of the force feedback term of the velocity command (5). The calculation time of the virtual environment is 1 ms/loop, which is small enough compared to the video frame rate (33 ms). The virtual arm and the massive payload from the four different viewpoints are displayed on a screen (see Fig. 7). Zooming the camera is also possible on the computer graphics. The motions of the virtual arms are displayed with a solid model. The motions of the slave arms constructed from the joint data transferred from space are superimposed with a wire frame model on the motion of the virtual arms. This makes it easier to check whether the slave arm follows the virtual arm after a time delay.

### Verification experiments

#### Validity of HIL simulation

HIL simulations were performed to confirm the validity of the system. In the HIL simulation, it is assumed that



**Fig. 7** Overview of virtual environment

the slave arm grips a massive payload of 3000 kg and pulls it along the x-axis at a speed of 0.01 m/s for 20 s. The mass and inertia of the virtual payload assumed in the experiment are presented in Table 1. The gains used in the experiment are shown in Table 2.

In the HIL simulation, the step input of the reference velocity of  $v_{s,r} = 0.01$  m/s was given instead of human teleoperation. The velocity command  $v_{s,c}$  was calculated by (2) with  $v_{s,r} = 0.01$  m/s.  $v_{s,r}$ ,  $v_{s,c}$ , and  $v_s$  during the HIL simulation are plotted in Fig. 8. As it satisfies this condition, the end effector velocity  $v_s$  is reduced by force feedback and asymptotically converges to  $v_{s,r}$  (0.01 m/s).

Figure 9 shows the force  $f_s$  measured by the force sensor at the wrist of the slave arm. In order to accelerate the massive payload of 3000 kg to a velocity of 0.01 m/s, a momentum of 30 Ns is necessary. As shown in Fig. 8, a step reference velocity  $v_{s,r} = 0.01$  m/s was given at  $t = 176.5$  s and  $v_s$  almost converged to  $v_{s,r}$  at  $t = 186$  s.  $f_s$  plotted in Fig. 9 was integrated from 176.5 to 186 s, and the integral value was 28.43 Ns. It is assumed that the velocity of the massive payload is almost the same as the velocity of the slave arm. Therefore the movement of the massive payload in the HIL

**Table 1** Mass and inertia parameters of the payload

Mass [kg]	Inertia [kgm <sup>2</sup> ]
500	diag [7.47 7.60 7.61]
3000	diag [44.82 45.61 45.64]

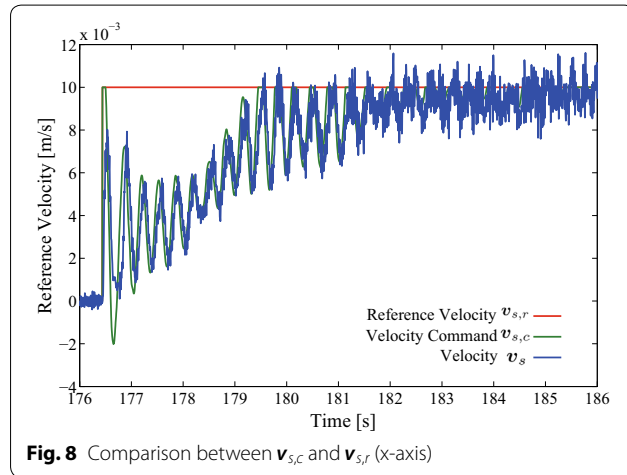
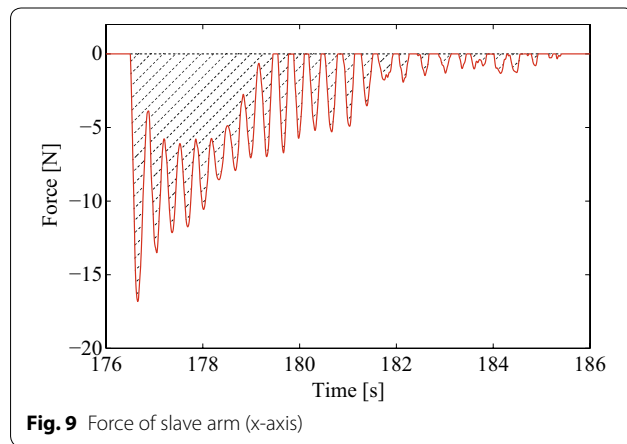
simulation almost agrees with theory. The error in the momentum required to accelerate the massive payload from 0 to 0.01 m/s was about 5%. The force feedback term of the control law of the slave arm adequately mitigated the applied force and the velocity of the slave arm asymptotically converged to the given reference velocity.

#### Comparison of the results with and without force feedback

Another HIL simulation was performed to verify the effectiveness of the force feedback in (2) in reducing the resultant force during the massive-payload manipulation mission. When a virtual payload of 3000 kg is manipulated in a HIL simulation without force feedback, a huge force will be generated between the payload and the end

**Table 2** Gain parameters of the experiment

$K_V$	diag [80/1000 80/1000 80/1000 0 0 0]
$K_S$	diag [1/1400 1/1400 1/1400 0 0 0]
$D$	diag [0.3 0.3 0.3 0 0 0]
$K_t$	diag [746,000 746,000 746,000 0 0 0]
$D_t$	diag [1000 1000 1000 0 0 0]

**Fig. 8** Comparison between  $v_{s,c}$  and  $v_{s,r}$  (x-axis)**Fig. 9** Force of slave arm (x-axis)

effector of the slave arm, and hence there is a risk that the slave arm used in this research will be broken. Therefore, in this HIL simulation, a massive payload of 500 kg is used instead of 3000 kg.

The massive payload of 500 kg was grasped by the slave arm and pulled automatically for 1 min at a speed of 0.01 m/s along the x-axis with and without force feedback. Figure 10 shows the force generated between the slave arm and the payload in an automatic manipulation experiment for the payload of 500 kg without force

feedback. In the experiment without force feedback, a maximum force of 23 N was generated when the payload was grasped, the arm started to pull the payload, and then the arm stopped pulling, as shown in Fig. 10.

Figure 11 shows the force generated between the slave arm and the payload in an automatic manipulation experiment of 500 kg with force feedback. In the experiment with force feedback, a maximum force of 18 N was generated at the time when the payload was grasped, the arm started to pull the payload, and then the arm stopped pulling, as shown in Fig. 11. During the manipulation, the generated force was kept small. The mean square error of  $f_s$  is calculated as follows to compare the results of the experiments with and without force feedback:

$$MSE(f_s) = \frac{1}{N} \sum_{t=0}^{\infty} \|f_s\|^2, \quad (11)$$

where  $N$  is the number of number of data items.  $MSE(f_s)$  was 5.41 N<sup>2</sup> and 15.35 N<sup>2</sup> when the force feedback was applied and was not applied, respectively.

Thus, it can be concluded that the force feedback in the slave arm controller helps reduce the force generated at the slave arm when the arm conveys a massive payload.

## HIL simulation of massive-payload manipulation

### Scenario of experiments

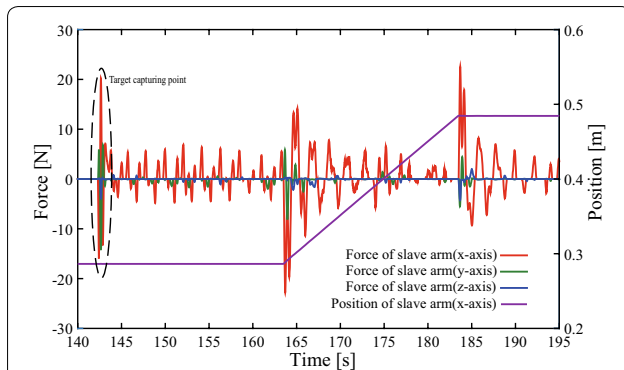
Two types of HIL simulations (HILS 1 and HILS 2) were performed.

First, a virtual massive payload of 4500 kg was grasped by the slave arm and was transferred along the x-axis at a constant speed of 0.01 m/s for 20 s (HILS 1). Next, a virtual massive payload of 4500 kg was grasped by the slave arm and was transferred along the x-axis by human teleoperation with a communication delay of 5 s (HILS 2) (see Additional file 1). In the teleoperation experiment, the input values for the y and z axes components are assumed to be zero.

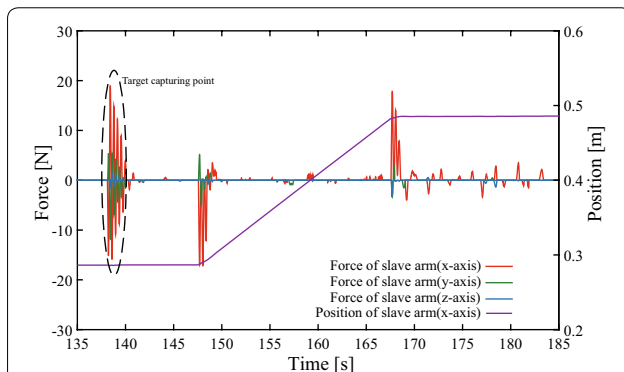
The effectiveness of the model-based teleoperation using the mixed force and motion commands for the problem of communication delay was verified in [6] by performing a teleoperation experiment between German Aerospace Center (DLR, Germany) and Tohoku University (Japan). In this work, the effectiveness of the modified version of the model-based teleoperation for massive-payload manipulation is verified.

Although the same HILSs were carried out with virtual payloads of 3000 kg and of 6000 kg, only the result with the virtual payload of 4500 kg is presented in this paper. The mass and inertia used in the HILS are shown in Table 3. The gain parameters of the experiment were





**Fig. 10** Force and position of slave arm without force feedback



**Fig. 11** Force and position of slave arm with force feedback

**Table 3** Mass and inertia parameters of the payload (4500 kg)

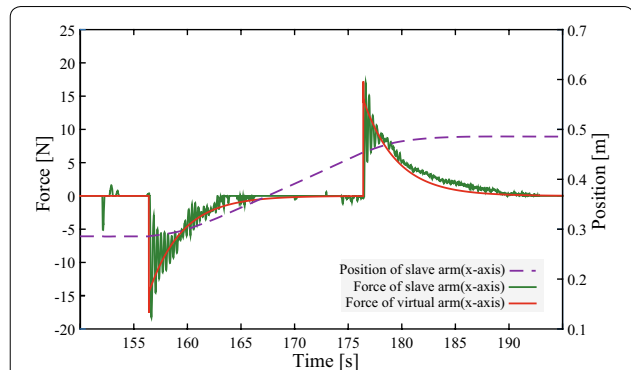
Mass [kg]	Inertia [kgm <sup>2</sup> ]
4500	diag [67.22 68.42 68.46]

the same as the parameters used in the experiments presented in “[Verification experiments](#)” section.

#### Comparison between the end effector force of the slave arm and the end effector force of the virtual arm (HILS 1)

In order to evaluate the accuracy of the simple virtual force model illustrated in Fig. 6, the end effector force of the slave arm generated along the x-axis is compared with that of the virtual arm as shown in Fig. 12.

Although the oscillation of the force was not reproduced by the simple spring-damper model, the model reproduced the mean value of the force with sufficient accuracy. It can be concluded that the dynamics of the massive payload of the slave side was modeled well by the simple spring-damper virtual force model.



**Fig. 12** Force of slave arm and virtual arm (x-axis) (in HILS 1)

#### Comparison between the end effector position of the slave arm and the end effector position of the virtual arm (HILS 1)

The positions of the end effectors of the slave and virtual arm along the x-axis are plotted in Fig. 13. Because the positions of both arms along the y and z axes were almost constant and identical, only the positions along the x-axis are presented.

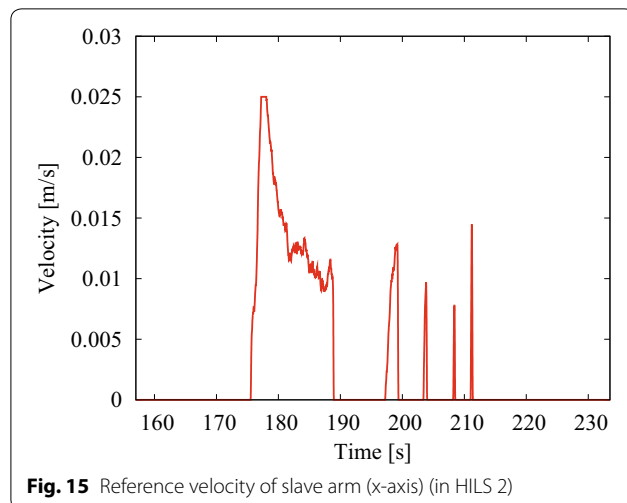
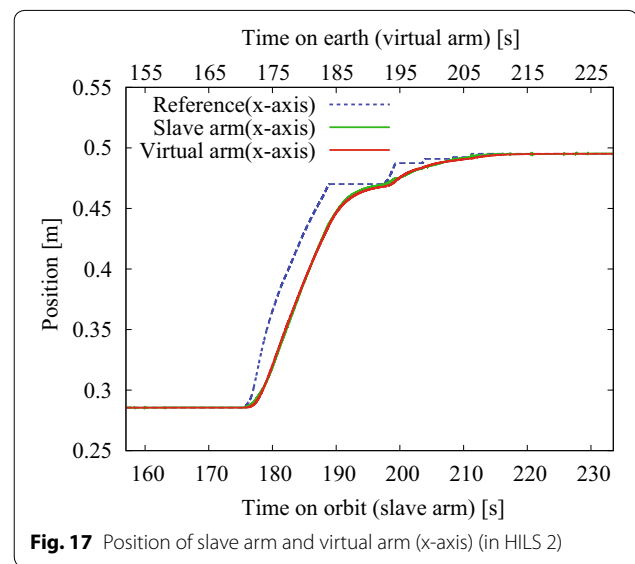
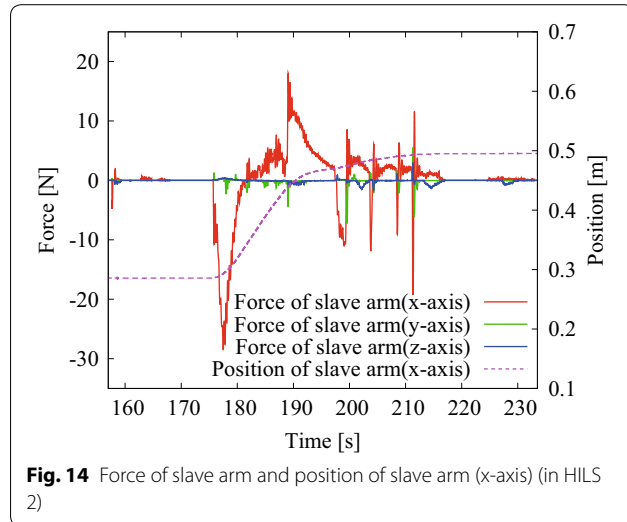
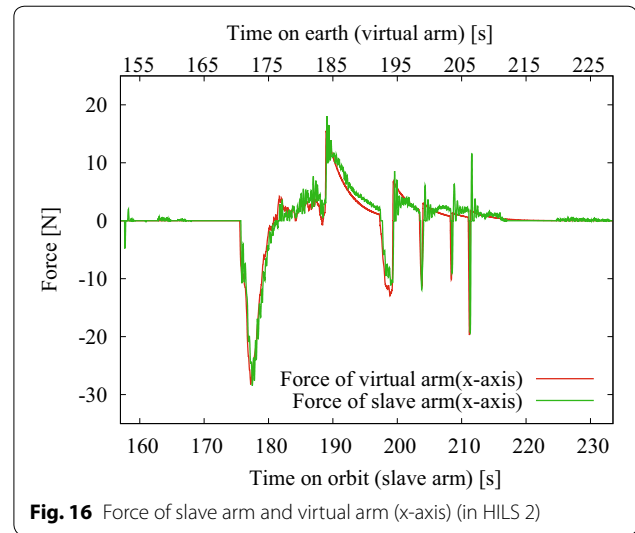
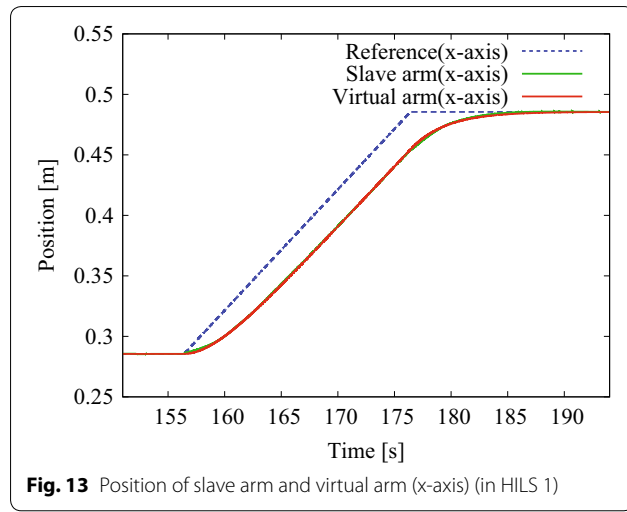
Because the position of the virtual arm was sent to the slave arm, and it was used in the slave arm control (the third term of the right hand side of 2), the motion of the virtual arm almost corresponds to the motion of the slave arm (note that the time delay was zero in HILS 1).

#### Comparison between the end effector force of the slave arm and the end effector force of the virtual arm (HILS 2)

Figure 14 shows the end effector force along the x, y, and z axes of the slave arm and the end effector position along the x-axis of the slave arm, respectively, when a virtual massive payload of 4500 kg was transferred by human teleoperation with a time delay (HILS 2). Figure 15 shows the reference velocity of the slave arm given by human teleoperation (see 4).

In HILS 2, the maximum end effector force was approximately 28 N. Compared to the result in HILS 1, the maximum end effector force generated in HILS 2 was larger, because the end effector reference velocity  $v_{s,r}$  generated by human teleoperation tends to be larger than the constant reference velocity  $v_{s,r} = 0.01$  m/s given in HILS 1 (see Fig. 15).

Figure 16 shows the forces generated at the slave and virtual arms. Note that the time axis on earth (for the virtual arm) is shifted 5 s later for the time axis on orbit (for the slave arm) in Figs. 16 and 17 in order to compare the results of the slave and virtual arms.



As shown in Fig. 16, the force generated at the virtual arm simulates the force of the slave arm well, although a small oscillation of the force was not reproduced.

#### Comparison between the end effector position of the slave arm and the end effector position of the virtual arm (HILS 2)

Figure 17 shows the positions of the end effectors of the slave and virtual arms. The maximum error between the positions of the arms was 0.002 m, which was slightly larger than the errors in HILS 1.

Thus, it is proved that the motion of the virtual arm predicts the motion of the slave arm well, and hence the

proposed method is effective for the massive-payload manipulation task by teleoperation with time delay.

## Conclusion

In this paper, a teleoperation system for a massive payload transportation task was proposed. The proposed control strategy was verified by performing hardware-in-the-loop simulations (HILSs).

It is confirmed that the control law to reduce the force generated between the slave arm and the massive payload works well. Furthermore, a virtual arm control law to reproduce the dynamics of massive payloads in a virtual environment is proposed. Several HILSs were carried out to verify the proposed system, and the differences between the results with and without the force feedback term were compared. It was verified that the massive-payload manipulation task can be performed while reducing the force generated between the end effector and the massive payload. HILSs of massive payload transport were also performed under a time delay environment, and it was verified that the movements and forces on the slave side were well reproduced in the virtual environment. The modified version of the model-based teleoperation will be applied to more complicated tasks in future work.

## Additional file

**Additional file 1.** This is a movie file of the massive-payload manipulation simulation experiment.

## Authors' contributions

AK led and directed the project. AK showed the need for dealing with massive payloads in teleoperation. JS proposed the control law to reduce the force generated between the end effector and the massive payload using the force sensor value of the slave arm. JS also proposed the control law of the virtual arm to reproduce the dynamics of massive payloads in the virtual environment. AK participated in the discussion on the optimization of the control law. All authors read and approved the final manuscript.

## Author details

<sup>1</sup> Graduate School of Information Science and Technology, Hokkaido University, Kita 14 Nishi 9, Kita-ku, Sapporo, Japan. <sup>2</sup> Department of Electrical Engineering, Shibaura Institute of Technology, 3-7-5, Toyosu, Koto-ku, Tokyo, Japan. <sup>3</sup> Graduate School of Engineering, Tohoku University, 2-1-1 Katahira, Aoba-ku, Sendai, Japan.

## Acknowledgements

The authors acknowledge Tohoku University for the experimental equipment and we are grateful Dr. Sugai for the robot programming support.

## Competing interests

The authors declare that they have no competing interests.

## Availability of data and materials

The datasets used and/or analysed during the current study are available from the corresponding author on reasonable request.

## Funding

The authors received no specific funding for this research.

## Appendix A: Dynamics of the virtual arm

If high-ratio reduction gears are employed at the joint, the joint angles and joint torques between before and after the gear reduction are related by

$$\mathbf{G}_r \mathbf{q} = \mathbf{q}_a, \quad (12)$$

$$\boldsymbol{\tau} = \mathbf{G}_r^T \boldsymbol{\tau}_a \quad (\because \Delta \mathbf{q}^T \boldsymbol{\tau} = \Delta \mathbf{q}_a^T \boldsymbol{\tau}_a = \Delta \mathbf{q}^T \mathbf{G}_r^T \boldsymbol{\tau}_a), \quad (13)$$

where  $\mathbf{G}_r$  is a diagonal reduction matrix, the diagonal elements of which correspond to the reduction ratio;  $\mathbf{q}$  is the joint angle vector of an arm;  $\mathbf{q}_a$  is the angle vector of the actuators;  $\boldsymbol{\tau}$  is the joint torque vector; and  $\boldsymbol{\tau}_a$  is the torque vector of the actuators.

The dynamics of the arm is expressed by

$$\boldsymbol{\tau} = \mathbf{M} \ddot{\mathbf{q}} + \mathbf{h} + \mathbf{J}^T \mathbf{F}, \quad (14)$$

where  $\mathbf{M}$  is the inertia matrix of the arm,  $\mathbf{h}$  represents the Coriolis force and centrifugal force and gravitational force,  $\mathbf{J}$  is the Jacobian matrix, and  $\mathbf{F}$  is the external force. Substituting (12) and (13) into (14), the following equation is obtained:

$$\boldsymbol{\tau}_a = \mathbf{G}_r^{-T} \mathbf{M} \mathbf{G}_r^{-1} \ddot{\mathbf{q}}_a + \mathbf{G}_r^{-T} (\mathbf{h} + \mathbf{J}^T \mathbf{F}). \quad (15)$$

Assuming that the actuator is the velocity command type, the actuator torque is calculated as

$$\begin{aligned} \boldsymbol{\tau}_a &= \mathbf{K}_{va} (\dot{\mathbf{q}}_{ca} - \dot{\mathbf{q}}_a) \\ &= \mathbf{K}_{va} \mathbf{G}_r (\dot{\mathbf{q}}_c - \dot{\mathbf{q}}) \quad (\because \dot{\mathbf{q}}_{ca} - \dot{\mathbf{q}}_a = \mathbf{G}_r (\dot{\mathbf{q}}_c - \dot{\mathbf{q}})), \end{aligned} \quad (16)$$

where  $\dot{\mathbf{q}}_{ca}$  is the joint angle velocity command.  $\mathbf{K}_{va}$  corresponds to the gain of the hardware velocity servo.

Substituting (16) into (15), the following equation is obtained:

$$\dot{\mathbf{q}}_c - \dot{\mathbf{q}} = \mathbf{G}_r^{-1} \mathbf{K}_{va}^{-1} \mathbf{G}_r^{-T} \mathbf{M} \mathbf{G}_r^{-1} \ddot{\mathbf{q}}_a + \mathbf{G}_r^{-1} \mathbf{K}_{va}^{-1} \mathbf{G}_r^{-T} (\mathbf{h} + \mathbf{J}^T \mathbf{F}). \quad (17)$$

By multiplying both sides of (17) by  $\mathbf{J}$ , the following equation is obtained:

$$\begin{aligned} \mathbf{J} (\dot{\mathbf{q}}_c - \dot{\mathbf{q}}) &= \mathbf{J} \mathbf{G}_r^{-1} \mathbf{K}_{va}^{-1} \mathbf{G}_r^{-T} \mathbf{M} \mathbf{G}_r^{-1} \ddot{\mathbf{q}}_a + \mathbf{J} \mathbf{G}_r^{-1} \mathbf{K}_{va}^{-1} \mathbf{G}_r^{-T} (\mathbf{h} + \mathbf{J}^T \mathbf{F}). \end{aligned} \quad (18)$$

Therefore, by using  $\mathbf{v}_c = \mathbf{J} \dot{\mathbf{q}}_c$  and  $\mathbf{v} = \mathbf{J} \dot{\mathbf{q}}$ , (18) is rewritten as

$$\begin{aligned} \mathbf{v}_c - \mathbf{v} &= \mathbf{J} \mathbf{G}_r^{-1} \mathbf{K}_{va}^{-1} \mathbf{G}_r^{-T} \mathbf{M} \mathbf{G}_r^{-1} \ddot{\mathbf{q}}_a \\ &\quad + \mathbf{J} \mathbf{G}_r^{-1} \mathbf{K}_{va}^{-1} \mathbf{G}_r^{-T} (\mathbf{h} + \mathbf{J}^T \mathbf{F}). \end{aligned} \quad (19)$$

If the reduction ratio is large enough, the two terms on the right hand side of (19) can be ignored, because the first term is proportional to  $\mathbf{G}^{-3}$  and the second term is

proportional to  $G^{-2}$ . In this case,  $\mathbf{v}$  can be approximated by  $\mathbf{v}_c$ . In massive-payload manipulation simulation,  $\mathbf{F}$  in the second term may become very large and hence it may not be possible to ignore the second term. However, if  $\mathbf{F}$  is adequately controlled by force feedback (e.g.  $\mathbf{K}_s \mathbf{F}_s$  in 2), the second term of the right hand side of (19) will be sufficiently small and the assumption  $\mathbf{v} \approx \mathbf{v}_c$  will hold.

## Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 12 December 2017 Accepted: 22 August 2018

Published online: 31 August 2018

## References

1. Cl J, Craig RC, David LA (2001) Time delay and communication bandwidth limitation on telerobotic control. *Int Syst Smart Manuf Int Soc Optics Photon* 4195:405–419
2. Secchi C, Stramigioli S, Fantuzzi C (2008) Variable delay in scaled port-Hamiltonian telemanipulation. *Mechatronics* 18(7):357–363
3. Sayers CP, Richard PP, Louis LW, Dana RY (1998) Teleprogramming for subsea teleoperation using acoustic communication. *IEEE Jo Ocean Eng* 23(1):60–71
4. Niemeyer G, Slotine JJE (2004) Telemanipulation with time delays. *Int J Robot Res* 23(9):873–890
5. Ryder CW, Sean MS, Elliot WH, David LC, Hao J, Mark RC, Allison MO (2014) Time-delayed teleoperation for interaction with moving objects in space. In: 2014 IEEE international conference on robotics and automation (ICRA)
6. Tsumaki Y, Goshozono T, Abe K, Uchiyama M, Koeppe R, Hirzinger G (2000) Verification of an advanced space teleoperation system using internet. In: Proceedings of the 2000 IEEE/RSJ international conference on intelligent robots and systems (IROS 2000), vol. 2. pp 1167–1172
7. Tsumaki Y (1998) A model-based space teleoperation system using mixed force and motion commands. *Trans JSME* 64(626):3901–3906
8. Yoon WK, Goshozono T, Kawabe H, Kinami M, Tsumaki Y, Uchiyama M, Oda M, Doi T (2004) Model-based space robot teleoperation of ETS-VII manipulator. *IEEE Trans Robot Autom* 20(3):602–612
9. Yoshida K (1996) Experimental platforms for research and development of space robots. *J Robot Soc Japan* 14–1:18–21
10. Wenfu X, Bin L, Yangsheng X, Cheng L, Wenyi Q (2007) A ground experiment system of free-floating robot for capturing space target. *J Intell Robot Syst* 48(2):187–208
11. Rouleau G, Rekleitis I, L'Archeveque R, Martin E, Parsa K, Dupuis E (2006) Autonomous capture of a tumbling satellite. In: Proceedings 2006 IEEE international conference on robotics and automation (ICRA 2006)
12. Boge T, Wimmer T, Ma O, Zebeay M (2010) Epos-a robotics-based hardware-in-the-loop simulator for simulating satellite rvd operations. In: 10th International symposium on artificial intelligence robotics and automation in space
13. Osaki K, Konno A, Uchiyama M (2010) Delay time compensation for a hybrid simulator. *Adv Robot* 24(8–9):1081–1098
14. Takahashi R, Ise H, Konno A, Uchiyama M, Sato D (2008) Hybrid simulation of a dual-arm space robot colliding with a floating object. In: 2008 IEEE international conference on robotics and automation (ICRA). pp 1201–1206
15. Uchiyama M, Iimura Ki, Pierrot F, Unno K, Toyama O (1992) Design and control of a very fast 6-dof parallel robot. In: Robotics mechatronics and manufacturing systems. pp 473–478
16. Tsumaki Y, Naruse H, Nenchev DN, Uchiyama M (1998) Design of a compact 6-dof haptic interface. In: Proceedings of the 1998 IEEE international conference on robotics and automation, vol. 3. pp 2580–2585
17. Kikuuwe R, Yasukouchi S, Fujimoto H, Yamamoto M (2010) Proxy-based sliding mode control: a safer extension of pid position control. *IEEE Trans Robot* 26(4):670–683
18. Kikuuwe R, Yamamoto T, Fujimoto H (2008) A guideline for low-force robotic guidance for enhancing human performance of positioning and trajectory tracking: it should be stiff and appropriately slow. *IEEE Trans Syst Man Cybern* 38(4):945–957

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)