Semi-Autonomous Control of Single-Master Multi-Slave Teleoperation of Heterogeneous Robots for Multi-Task Multi-Target Pairing

Yushing Cheung
Department of Mechanical Engineering,
Stevens Institute of Technology,
Hoboken NJ 07030, USA
johnyscheung@gmail.com

Jae H. Chung
US Army RDECOM-ARDEC,
Building 95N,
Picatinny Arsenal, NJ 07806, USA
jchung3@stevens.edu

Abstract

This paper focuses on developing a control method for a single-master multi-slave (SMMS) system to cooperatively control a team of mobile agents for multi-robot multi-task multi-target (MRMTMT) pairing. Major components of the developed control method are (1) modified potential field based leader-follower formation, (2) adaptive master-slave impedances, (3) compensation for human induced errors and contact forces, and (4) robot-task-target pairings. Component (1)-(3) have been developed for formation reconfiguration, collision avoidance, and motion control for reliability and robustness.

Component (4) is derived from proven auction algorithms for MRMTMT cases, which optimizes robot-task-target pairing based on heuristic data. The MRMTMT pairing method is developed using a weighted attack guidance table (WAGT), which includes benefits of subteam-robot-task-target pairs. It converges rapidly - as is the case for auction algorithms with integer benefits. Simulations illustrated efficacy of a SMMS system with proposed Component (1)-(4) of the control method for multi-target capturing and transporting.

Keywords: Teleoperation, heterogeneous robots, multi-agent, and multi-target operations

1 Introduction

Cooperative control of multi-robotic systems has been studied extensively in recent years [9, 16, 22], especially for some tasks that cannot be handled by one single robot. It can improve dexterity of robots and enlarge application fields of robots. Thus, many cooperative control algorithms have been proposed so far [9, 16, 22]. There are two types of cooperation. One is the cooperation without force interactions among robots (unconstrained motion tasks) and the other is with them (constrained motion tasks). Besides these two types of cooperation, the transition between them has been investigated in some papers [16]. The transition involves a smooth, stable switching between motion and force control when instability and large force spikes during the switching are avoided. For the unconstrained motion tasks, e.g. searching in an open area [9], Fong, Thorpe, and Baur et. al. suggested the collaborative control with dialogue functions to remotely operate the multi-robot via a master robot when for constrained motion tasks, e.g. an object transportation [22], Yamamoto and Fukuda et. al. proposed a multi-robot path trajectory planner. However, in the papers [9, 16, 22], it was not shown that any control method was developed to enable robots to do the unconstrained and constrained motion tasks altogether when some applications, such as military operations, space explorations, and etc, require them to execute the multi-motion tasks at the same time.

Furthermore, in most of the applications, unstructured nature of the worksite environments and the limitations of the current sensors and computer decision-making technologies prohibit the use of fully autonomous systems for the operations [1, 6, 8, 10, 11, 12, 15, 17]. Therefore, it is required that the human decision making be involved in the systems. Teleoperators, in which a human operator is an integral part of the control, are established to integrate the human decisions to the control loop of the systems. The teleoperators for a cooperative multi-robotic system are classified into two types of the systems regarding the number of master robots, Single-Master Multi-Slave System (SMMS) and Multi-Master Multi-Slave System (MMMS). In order to minimize the required human resources and amplify the human effort, only the SMMS teleoperation has been considered in these papers [8, 10, 11, 12, 17].

Moreover, due to the complexity of the applications [20, 21], heterogeneous robots could be involved in doing a multi-task mission. For example, an unmanned aerial vehicle is teamed up with several unmanned ground vehicles. The unmanned aerial vehicle is used to survey an uncertain area while the unmanned ground vehicles being supervised by the unmanned aerial vehicle are moving into the uncertain area and doing an operation. A pairing method for the heterogeneous robots has become important for effectiveness of the mission accomplishment. Some heterogeneous multi-robot pairing methods have been proposed in the papers [2, 7, 14, 13]. Nonetheless, they did not produce a stable and fast convergence to a global optimum [7, 14, 13] and could not be used to pair different robots, tasks, and targets [2].

The primary objective of this paper is to develop a control method for a SMMS system to cooperatively control heterogeneous mobile robots for multi-robot multi-task multi-target (MRMTMT) pairing. Primary components of the proposed control method are (1) modified potential field based leader-follower formation, (2) compensation for operator-induced errors and contact forces, (3) adaptive impedances of master-slave robots [3, 4, 5], and (4) MRMTMT pairings. The primary components from (1) to (3) have been developed in our paper [3, 4, 5]. During the operation, a human operator only concentrates on controlling the team leader robot. All other team robots autonomously make a formation with regard to its positions and velocities based on sensory information [3, 4, 5]. Moreover, the compensator for human induced errors developed in [3, 4, 5] is used to regulate the transparency regarding the human induced error detections to make the SMMS system free of being affected by inconsistent human performance. As soon as the team leader is close enough to the multi-target locations, the team is autonomously split into several subteams, and each subteam leader is online selected based on its robot functionalities and proximity to an assigned target to lead the subteam. Each subteam leader supervises the followers of the subteam to execute the assigned task on the assigned target when the subteam followers move with respect to their subteam leader motion. The subteam can avoid obstacles and approach the assigned target in a modified potential field based leader-follower formation. With the adaptive impedances of master-slave robots and compensation for contact forces, the subteam robots are enabled to perform different motions, e.g. unconstrained, constrained, and transition ones, locally for various targets.

The rest of this paper is organized as follows. In Section 2, a control method is developed to integrate the primary components to control multi-subteams to execute multi-tasks simultaneously for a multi-target approach. In Section 3, the conditional transparency [3, 4, 5], i.e. the transparency if no human induced error is found, and effectiveness of the task achievement of the SMMS teleoperation system with the proposed control method are evaluated through simulation studies. Section 4 concludes this paper and discusses future research.

2 Semi-autonomous teleoperation control method for a multi-robots-multi-targets approach

This paper extends the preliminary concepts of the semi-autonomous single-master multi-slave teleoperation control method [3, 4, 5] which was only focused on a single target operation into a multi-target operation, i.e. several simultaneous target captures and transportations, in complex environment. The primary difference between the team robot coordinations to complete multi-tasks for multi-targets and a single task for a single target is multi-robot/multi-task/multi-target (MRMTMT) pairings.

The MRMTMT pairing method is developed to advance the semi-autonomous teleoperation control method that was seen in our papers [5, 4, 3] to form subteams to perform assigned tasks on assigned targets. The concept of the MRMTMT pairing method is that all robots act largely independently in terms of planning for themselves but are able to take into account team resources by working on the tasks with other team members. With the MRMTMT pairing method developed in this paper, the team leader not only takes any human command from the master robot but also works as an auctioneer to send and show all bid data e.g. target locations and their base prices that are also online shared by all other robots called team followers. A team robot is online appointed as the team leader by the human operator via the master robot if the original one [3, 4, 5]. All team followers act as bidders to form a subteam by themselves in order to maximize a sum of all follower bid values and bid on the targets when the corresponding task on the targets is performed by the cooperation of the subteam. In the subteam, the bidder with the maximum bid value is selected as a sub-team leader or a sub-leader, that is responsible for monitoring and coordinating all subteam member actions. According to the largest bid proposed by the subteam, the auctioneer, the team leader, decides which sub-team wins the bid with the restriction that each task and target are only gained by one subteam per auction. If all subteam bid values are smaller than the base price, or any slave robot can not compute the bid value due to lacking of the information surrounding the targets, the auctioneer obtains the bid. If any one of the subteams of the bidders already completes the task on the target, it will inform the auctioneer to cancel the bid.

After being assigned to the task and target by using the proposed pairing method, the robots in the subteam will move to the targets with the modified potential field based leader-follower formation control method and perform the task. If the assigned task is to request the assigned subteams to move in transition and constrained motions, e.g. pushing the target from place to place, the subteam impedances are adapted, and the local force compensator in the subteam is used to regulate the grasping force against the target to have a firm grip.

2.1 Multi-robot/multi-task/multi-target(MRMTMT)pairing method

Consider such a scenario, in a two-dimensional and limited rectangular environment X with n_c square cells, n_p slave robots pursue n_e targets, for $n_p > n_e$. The set of the robots is denoted by a matrix of $A = [a_1, a_2, ...a_{n_p}]$ where a_{n_p} is a robot matrix of n_p. Robot Capability Vector j for Task t is denoted by $\hat{C}_j^t$, $1 \leq j \leq n_p$, and the set of targets is represented by a target matrix of $T = [T_1, T_2, ...T_{n_e}]$ where T_{n_e} is a target matrix of n_e. The vector representing the capability required to accomplish Task t on Target T is denoted by $\bar{C}_t^T$, $1 \leq T \leq n_e$. Agent $A \cup T$ denotes robot teams and targets. For simplification, we assume that both space and time can be quantized, therefore the environment can be regarded as a finite collection of cells, denoted by $X_c = 1, 2, ..., n_c$. There exist some static obstacles with fixed sizes and regular shapes, and their locations are determined by the mapping $m : X_c \rightarrow 0, 1$, for $\forall x \in X_c, M(x) \geq thresh1$ indicates that the cell x is occupied by obstacles. $\forall x \in X_c, M(x) \leq thresh2$ indicates that the cell x is free, where $thresh2 < thresh1$ represents the threshold value between 0 and 1. Thus, each robot has different capabilities to complete different tasks on different targets.

2.1.1 Robot capability: For Task t and Robot j, the weighted capability vectors of Robot j can be defined as

$$\hat{C}_j^t = w_j^T diag\{b_{j1}^t, b_{j2}^t, ..., b_{ju}^t\} \begin{bmatrix} c_{j1}^t & \ldots\ldots & c_{ju}^t \end{bmatrix}^T \tag{1}$$

where u is the maximum number of the vectors, each of which represents the individual functionality. The set of robot matrices is rewritten into $A = \begin{bmatrix} a_{11} & \ldots\ldots & a_{1r} \\ a_{21} & \ldots\ldots & a_{2r} \\ \ldots\ldots & \ldots\ldots & \ldots\ldots \\ a_{n_v1} & \ldots\ldots & a_{n_v} \end{bmatrix}$ where n_v, for

$0 < n_v \leq n_p$, is the total number of the robots in the team, and r, for $0 < r \leq n_e$, is the total number of the tasks. c^t_{jk} is a capability vector for Functionality k and Task t. w^T_j is a positive integer such that for Target T and Robot j, the following is satisfied. If the robot is assigned to the target, $w^T_j = 0$, otherwise, $w^T_j = 1$. The $u \times u$ dimension diagonal matrix of b^t_{ju} is used to estimate the percentage of possibility of using the $u \times 1$ dimensional capability vector C^t_j to do Task t by Robot j successfully. However, if Robot j does not have Capability c^t_{jk}, then the b^t_{jk} is 0. Each robot matrix in A has weighted capability vectors, e.g. for Robot j and Task t, $a_{jt} = [\,\hat{C}^t_j\,]^T$.

2.1.2 Capability required to execute tasks on targets: It is assumed that there are p tasks which need to be done independently and simultaneously. All tasks are represented by a set of task matrices e.g. $t = \{t_1, ...t_p\}$ in the system for $p \leq n_e$, i.e. one task can be paired to two or more targets, but each target can only be paired to one task. The capability vector that is required to accomplish Task t on the Target T is defined as

$$\bar{C}^T_t = diag\{\beta^T_{t1}, \beta^T_{t2}, ..., \beta^T_{tu}\}C_{tu} \tag{2}$$

where the $u \times u$ dimension diagonal matrix of β^T_{tu} is used to describe the percentage of possibility of using the $u \times 1$ dimension capability vector C_{tu} with which the robot can finish Task t on Target T. $C_{tu} = [c_{t1}......c_{tu}]^T$ when the total number of the vectors of the functionalities is u. c_{tu} is the capability vector that is required to complete Task t with Functionality u. However, if Task t cannot be done successfully by any robot with the capability C_{tu} on Target T, then the β^T_{tu} is 0. Otherwise, β^T_{tu} is 1. Otherwise, β^T_{tu} is 1.

2.1.3 Subteam capability: The subteam is a combination of the multi-robots that work on Task t cooperatively. For Robot j and Task t, $U_{(j,t)} = a_j e_t$ where e_t is one if Task t is assigned; otherwise, it is zero, and a_j is defined in Eq. (1) for $a_{max} \geq j \geq a_{min}$, $a_{min} \geq 1$, and $a_{max} \leq n_p$ where $n_p/(a_{max} - a_{min} + 1) = n_s$ where n_s is the total number of subteams, and a_{max} and a_{min} are the number of the first and last robots forming Subteam y, respectively. Subteam y is represented by a matrix of $D_y = \begin{bmatrix} U_{(a_{min},1)} & \cdots\cdots & U_{(a_{max},1)} \\ U_{(a_{min},2)} & \cdots\cdots & U_{(a_{max},2)} \\ \cdots\cdots & \cdots\cdots & \cdots\cdots \\ U_{(a_{min},r)} & \cdots\cdots & U_{(a_{max},r)} \end{bmatrix}$.
Then, matrix A denoting a robot team formed by subteams, one of which is represented by D_y, is rewritten into $A = \{D_1, .., D_y, .., D_q\}$ where q is the total number of the combinations of multi-robots (robot subteams) in the team. For Robot j and Task t, if $\hat{C}^t_j > 0$, then

$$Q_{(j,t)} = \hat{C}^t_j \quad for \quad n_p \geq j \geq 1 \tag{3}$$

where $Q = [\,Q_{(1,t)} \cdots Q_{(n_p,t)}\,]$ is a positive integer. Subteam y capability vector for Task t is defined as

$$\tilde{C}^y_{(y_a:y_b,t)} = \sum_{j=y_b}^{j=y_a} Q_{(j,t)} \tag{4}$$

where $y_b - y_a$, $\forall y_b \geq y_a$, is the total number of the robots in Subteam y. y_a is the first and y_b is the last indices of the elements in the matrix of Q for Task t and Subteam y. Subteam y is able to perform Task t on Target T if the condition, $\bar{C}^T_t \leq \tilde{C}^y_{(y_a:y_b,t)}$, is satisfied. Robot j is selected as a subteam leader when its magnitude of the capability vector $\hat{C}^t_j$ is largest in the same subteam. It is assumed that the subteam leader knows all capability information about its subteam members.

Table 1: Weighted Attack Guidance Table (WAGT)

Subteam 1	$m_{N,1}$	$\cdots$	Subteam n	$m_{N,n}$
$B^1_{11}, \ldots, B^1_{x1}$	$m_{1,1}$	$\cdots$	$B^1_{1n}, \ldots, B^1_{xn}$	$m_{1,n}$
$B^2_{11}, \ldots, B^2_{x1}$	$m_{2,1}$	$\cdots$	$B^2_{1n}, \ldots, B^2_{xn}$	$m_{2,n}$
$\cdots\cdots\cdots$	$\cdots$	$\cdots$	$\cdots\cdots\cdots$	$\cdots$
$B^N_{11}, \ldots, B^N_{x1}$	$m_{N,1}$	$\cdots$	$B^N_{1n}, \ldots, B^N_{xn}$	$m_{N,n}$

2.1.4 Bidding winner determination: In Table 1, $m_{N,n}$ is a positive integer weight for Subteam n to bid on Target N. If $\tilde{C}^n_{(y_a:y_b,x)}$ is smaller than the base price which is a positive integer, or Target N has already been assigned to Robot Subteam n, $m_{N,n}$, a positive integer, is 0. Otherwise, $m_{N,n}$ is 1. By arranging $m_{N,n}$ and B^N_{xn} into Table 1, called Weighted Attack Guidance Table (WAGT), each row of WAGT corresponds to a target and Robot Subteam (1 to n) when n is the total number of the subteams forming the team. In addition, each column of WAGT corresponds to a robot combination (Robot Subteam) that works on Targets (1 to N) when N is the total number of the targets. Therefore, there are the N rows and n columns in WAGT. The scanning proceeds from the first to the last column. Hence, the robot combination (Robot Subteam) specified in column i takes precedence over combination of robots specified in column $i+1$. For example, for Subteam n, Task t, and Target N, the bid value is weighted as follows.

$$B^N_{tn} = (\tilde{C}^n_{(y_a:y_b,t)} - \bar{C}^N_t)(1 - X^N_{tn}) \tag{5}$$

X^N_{tn} is the positive integer weight for Subteam n to do Target N. If Task t is the most preferred by Subteam n to be done on Target N when B^N_{tn} is the maximum value of the element in $\tilde{B}(N,n)$ where $\tilde{B}(N,n) = [\, B^N_{1n} \,\ldots\, B^N_{tn} \,\ldots\, B^N_{xn} \,]$ for Task $1, ..t, ..x$ and x is the total number of the tasks, then $X^N_{tn} = 0$; otherwise, $X^N_{tn} = 1$. Therefore, based on the given subteams, targets, tasks, WAGT, and optimization of the robot-target pairing that is described below, the bidding winner determination is made.

The optimization of the robot-task-target pairing is formulated as follows. Given Subteams $1 - n$, Targets N, Tasks $1 - t$, and WAGT, an assignment of the subteam is found in such a format that WAGT is produced as seen in Table 1, and its Objective Function $ObjFun(N)$ in Eq. (6) is maximized within the given constraints in Eq. (7) where $ObjFun(N) = \begin{bmatrix} (B^N_{11} m_{N,1}) & \ldots & (B^N_{1n} m_{N,n}) \\ \ldots & \ldots & \ldots \\ (B^N_{t1} m_{N,1}) & \ldots & (B^N_{tn} m_{N,n}) \end{bmatrix}$
Maximize

$$ObjFun(N) \tag{6}$$

Subject to

$$\sum_{y=1}^{y=n} \tilde{B}(N,y) \geq 0 \tag{7}$$

where $m_{N,n}$ is the positive integer weight for Subteam n and Target N. Initially, all $m_{N,n}$ is equal to one if no subteam is assigned to any target. However, if Subteam n is assigned to Target N, $m_{N,n}$ is equal to zero. Hence, Subteam n that proposes the maximum affordable value $(B^N_{tn} m_{N,n})$ can win Target N by solving Eqs (6) within the constraints in Eq. (7).

By using the MRMTMT pairing method, the subteam-task-target pairs are stored into the resulted matrices e.g. subteam-task-target pair matrix and given WAGT. For instances, Subteam n is paired to Task t and Target N when Subteam $(n+1)$ is paired to Task $(t+1)$ and Target $(N+1)$. The subteam-task-target pair matrix, J_I is written as $J_I = \left[\begin{smallmatrix} t & 0 \\ 0 & t+1 \end{smallmatrix}\right]$ when the first and second columns of the J_I represent Subteam n and $(n+1)$, respectively, and the first and second rows of the J_I represent Target N and $N+1$, respectively. As described above, the MRMTMT pairing method is created based on the found pairs to form subteams, appoint robots as subteam leaders and followers, pair tasks to the subteams, and generate position and/or force references to the subteams to work on the given targets.

The control method in [5] has been developed for a SMMS system to coordinate robots in a single-task single-target operation. Therefore, in this paper, in order to be able to split its slave team into subteams for a multi-task multi-target operation, the control method in [5] is modified by incorporating the mentioned MRMTMT pairing method. The other components of the proposed SMMS system are similar to those we have developed in [5].

2.2 SMMS teleoperation systems with the proposed modifications

In [5], the SMMS system with developed control method was used to coordinate slave robots in a single-task single-target operation. However, because most real applications are complicated, the SMMS system is required to have a capability of handling a multi-task multi-target mission. Therefore, the SMMS system is modified into the one with integrating the MRMTMT pairing method and the other components developed in [5] as shown in Figure 1. Figure 1 represents the overall architecture of the modified teleoperation system. The master and slave subsystems in Figures (1a) and (1b), respectively, are connected over the wireless internet. The difference from the one in [5] is that the slave subsystem with the proposed control methods is operated fully autonomously for two reasons. (1) Human commands via the master subsystem are temporarily not available due to intermittently disrupted or delayed transmission between the subsystems. (2) The team formed by the slave robots is required to be divided into subteams to simultaneously perform the task on the target when the robots in the subteam are successfully assigned to the proper tasks and targets with the MRMTMT pairing method. The modified system shown in Figure 1 can be formulated into the following equations of motion.

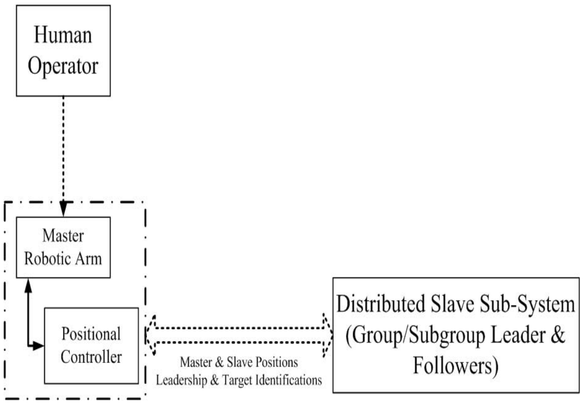
Master:

$$M_m \ddot{e}_m + B_m \dot{e}_m + K_m e_m = 0 \tag{8}$$

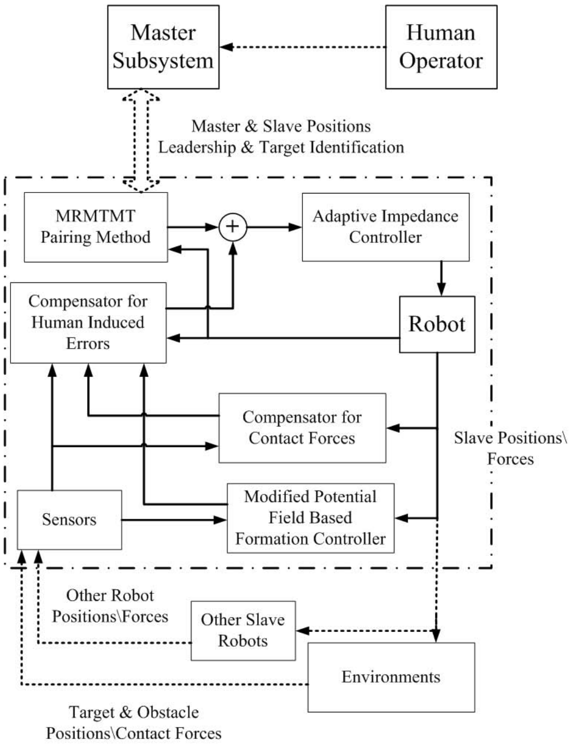
i^{th} Slave:

$$M_{si}\ddot{e}_{si} + B_{si}\dot{e}_{si} + K_{si}e_{si} = (1-W)(U_T + U_o) + (1-\sigma)(1-\lambda)U_f + v_{si}^T \tilde{a}_{si} + C_e \delta F_{si} \tag{9}$$

where U_f is the virtual bonding between robots. U_T is the virtual attraction to the target while U_o is the virtual repulsion from the obstacles. C_e is the force compensator to regulate the contact force acting against the target to make a firm grip. W is the sensor based compensator to adjust the transparency of the SMMS system based on the human induced error detection. U_f, U_T, U_o, C_e, and W were proposed in our paper [5]. $v_{si} = \left[\begin{smallmatrix} z_{si} & \dot{x}_{si} & x_{si} \end{smallmatrix}\right]^T$, $z_{si} = \mu(\sigma \ddot{x}_m + (1-\sigma)\ddot{x}_{sdi}) - \beta_1 \dot{e}_{si} - \beta_0 e_{si}$, β_1 and β_0 are positive constants chosen such that $s^2 + \beta_1 s + \beta_0$ is a stable (Hurwitz) polynomial when s is the Laplace operator. $\tilde{a}_{si} = W\left[\begin{smallmatrix} \hat{M}_{si}-M_{si} & \hat{B}_{si}-B_{si} & \hat{K}_{si}-K_{si} \end{smallmatrix}\right]^T$. $\hat{M}_{si}$, $\hat{B}_{si}$, and $\hat{K}_{si}$ are the estimated slave impedance matrices. x_m and x_{si} are the master and the i^{th} slave robot position vectors, respectively.



(a) Master subsystem



(b) Slave subsystems (leader/followers)

Figure 1: Modified SMMS Systems

x_{sdi} is the reference position vector of the i^{th} slave robot. M_m is the inertia matrix of the master robot. K_m is the control parameters for the linear diagonal master matrices. M_{si} is the inertia matrices of the i^{th} slave robots. B_{si} is the slave impedance matrix. K_{si} is the control parameters for the linear diagonal slave matrices. σ and λ are the control parameters of the i^{th} slave robot. When the robot is selected as a team leader, σ becomes one; otherwise, it becomes zero. When the robot is appointed as a subteam leader, λ becomes one; otherwise, λ is zero. B_m is the master adaptive impedance matrix. $e_{si} = x_{si} - (\sigma x'_m + (1-\sigma)x'_{ideal})(\alpha_1 + (1-\alpha_1)\psi_{pos})$. x_{si} is the slave current robot positions. $\alpha_1 = |sgn(e^2_{si})|$, which is the constant positive integers switching between zero and one in order to determine the output of the target matrix. ψ_{pos} is the matrix, $[\,0\;0\;1\,]^T$ to produce its reference position vectors transformed from X'_{ideal}. x'_m and x'_{si} are the delayed transmitted x_m and x_{si}, respectively. x_{ideal} is the slave subteam robot reference position vectors that can be generated from X_{ideal} $\delta F_{si} = F_{si} - F_{ideal}(1-\alpha_1)\psi_{force}$ is the difference between reference and measured forces of the slave robots when F_{ideal} is the reference force vectors that can be generated from X_{ideal} in and F_{si} is the measured forces of the slave robots. ψ_{force} is ,$[\,0\;1\;0\,]^T$, the matrix to produce its reference force vectors transformed from F'_{ideal}. The slave team leaders path remotely controlled by the human operator is adapted for good tracking of the master robot positions unless the path is too close to an obstacle and/or too far from the target. In other words, the team leader's transparency is compromised. Otherwise, it is adaptively enhanced. When the team is close enough to the region full of targets, the team becomes autonomous and can be split into subteams paired to tasks and targets by using the MRMTMT pairing method to solve Eqs (6) within the constraints in Eq. (7). Target and Task Matrices are generated and transformed into the reference positions and forces for the robots to accomplish the assigned tasks on the assigned targets. During navigation to the targets, due to no contact force, $\delta F_{si} = 0$, only the robot-obstacle, robot-target, and robot-robot distances can be sensed. If there are excessively robot-target distances and/or short robot-obstacle distances, all slave team robots could autonomously adjust their routes to adapt the formation to approach the target and avoid the obstacles. Furthermore, the subteam leader-follower formation can be maintained or distorted by integrating (1) the virtual robot-robot bondings with different strengths based on which two team robots are connected, (2) the attraction to the target with regard to robot-target distances, and (3) the repulsion from the obstacles with regard to robot-obstacle distances. In such a formation, all followers in the subteam move with regard to the subteam leader's motion. After the target is reached, the slave robots will perform the assigned tasks, such as target capture or transportation relying on the task and target matrices. During target transportation, the contact force against the target by each subteam robot is adjusted, which could cause the subteam robots to have a firm grip of the target while the target is being moved from place to place.

3 Simulation results

The target scenario for the simulation study is that mobile robots are required to form groups to handle a multi-target mission in the presence of time-varying communication delays, the proposed control method, as described in Table 2. In the simulation, the time dependent communication delays were simulated as shown in Figure 3. The maximum communication delay of 0.1 second was chosen in the simulations because for earth applica-

Table 2: SMMS simulations for a multi-target mission

Simulations	Robot types	Control objectives
Sim (1)	Homogeneous robots	MRMTMT pairing
Sim (2)	Heterogeneous robots	MRMTMT pairing

tions, there is a critical value, beyond which the system tends to become unstable [18]. The master robot is a joystick connected to a laptop that reads motion commands from a human operator.It was able to transmit motion commands to a virtual slave robot model over the simulated TCP/IP internet. The virtual slave robots shown in Figure 2(b) including a team leader, subteam leaders, and followers are modeled in Matlab software. The slave robots were simulated as holonomic mobile platforms with or without manipulators atop. Six virtual static obstacles and two virtual targets were placed in a virtual environment.

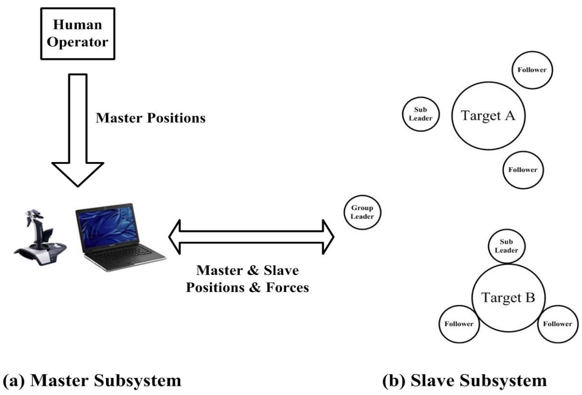


Figure 2: SMMS Teleoperation Simulation Setups

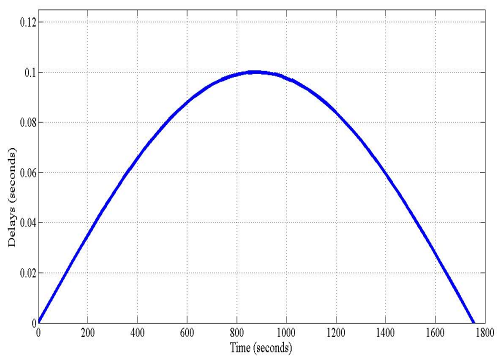


Figure 3: Time Delays in Simulations

All target and robot positions are assumed to be known and all obstacles are uncertain in the simulation, and the obstacles and targets are modeled as simple mass-spring-damper systems [18] as following. Targets:

$$M_T\ddot{x}_T + B_T\dot{x}_T + K_T x_T = F_T \tag{10}$$

Obstacles:

$$M_o\ddot{x}_o + B_o\dot{x}_o + K_o x_o = F_o \tag{11}$$

where M_T and M_o are the inertia matrices of the targets and obstacles, respectively. B_T and B_o are the damping coefficients of the targets and obstacles, respectively. K_T and K_o are the stiffness coefficients of the targets and obstacles, respectively. x_T and x_o are the position vectors of the targets and the obstacles, respectively. In the simulations, the seven slave robots were run to approach the two targets, Target A (TA) and B (TB), while getting around the seven static obstacles. The targets are initially stationary. The two simple tasks, transportation and capture, were performed by the slave robots simultaneously. TB required at least three mobile robots to be transported from place to place when TA was also captured by at least three mobile robots. TB was placed on a movable platform with four passive omni-directional wheels tightly touching the ground. There was no slip between the surfaces of the ground and the wheels. TB was transported only if it was being grasped since the slave robots surrounded and then pushed on it. TB would be damaged if the slave robots pushed too hard; however, the slip between the surfaces of TB and the slave robot arms would occur if they pushed too lightly. Besides, TA was fixed on the ground. It was being captured while being encircled by the slave robots.

The simulations were set up with the following parameters. The desired distance between two robots was $3m$. The minimum robot-obstacle distance was $5m$. Six circular objects with the radii of $5m$ were used as obstacles in each simulation. In the simulations, the six circular obstacles, Ob1, Ob2, Ob3, Ob4, Ob5, and Ob6, were situated at (30, 60), (50, 40), (70, 20), (70,-20), (50,-40), and (30,-60), respectively. Two circular objects with the radii of $5m$ represented TA and TB in the simulations. TA and TB were initially situated at (90, 30) and (90, -30), respectively. The seven slave robots, R1, 2, 3, 4, 5, 6, and 7, were initially located at (0, 15), (0, 10), (0, 5), (0, 0), (0, -5), (0, -10), and (0, -15), respectively. Only two directions parallel to the ground were considered in the simulations. Each slave robot was represented by a circular object with the radius of $3m$ in simulations. The master and slave forces and positions were simulated through the PC device and divided by 10, respectively. The TB were carried by slave robots from (90, -30) to (130, -30). In the simulations, the following parameters were used:
M_m = 3 kg, K_m = 6 Ns/m, M_{si} = 30 kg, B_{si} = 1.0 Ns/m, K_{si} = 60 N/m, M_T = 60 kg, B_T = 0.0 Ns/m, K_T = 800 N/m, M_o = 6000 kg, B_o = 0.0 Ns/m, K_o = 1000 N/m, $/mu$ = 10, k_e = 100, b_e = 60, r_{imin} = 5, r_{smin} = 5, k_f = 1, $\alpha = \rho = 1$, β_1 = 10000, β_0 = 500, ϕ = 100, and $\Lambda_i = \varphi = \gamma = \gamma_w$ =1

In the simulations, no friction, gravity, and air resistance were assumed in the virtual environment. If the contact forces exceeded 15 N, the target and robots were assumed to be damaged. The slip was programmed to occur between the contact surfaces of robotic tip point and the target surface only if the static friction condition [19] was not met, i.e. pushing force larger than the maximum allowable static friction where the friction coefficient was 0.5 [19]. In the simulations, the slave forces are defined as the contact forces sensed from the slave robots. The position errors are e_m and e_{si}. The force errors are δF_{si}. The simulations, Sim (1) and (2), as listed in 2 were conducted by the same operator for consistency. All slave robots were programmed to move at an average speed of 0.1 m/s in the virtual environment in order to evaluate the effectiveness of the proposed systems by measuring the length of time taken to complete those tasks. In Table 3, those robots could form subteams Sub1 - Sub35 in order to transport TB while capturing TA in the

Table 3: Robot Combinations (Robot Subteams)

Subteam	Combo	Subteam	Combo	Subteam	Combo
Sub1	R1 R2 R3	Sub13	R1 R5 R6	Sub25	R2 R6 R7
Sub2	R1 R2 R4	Sub14	R1 R5 R7	Sub26	R3 R4 R5
Sub3	R1 R2 R5	Sub15	R1 R6 R7	Sub27	R3 R4 R6
Sub4	R1 R2 R6	Sub16	R2 R3 R4	Sub28	R3 R4 R7
Sub5	R1 R2 R7	Sub17	R2 R3 R5	Sub29	R3 R5 R6
Sub6	R1 R3 R4	Sub18	R2 R3 R6	Sub30	R3 R5 R7
Sub7	R1 R3 R5	Sub19	R2 R3 R7	Sub31	R3 R6 R7
Sub8	R1 R3 R6	Sub20	R2 R4 R5	Sub32	R4 R5 R6
Sub9	R1 R3 R7	Sub21	R2 R4 R6	Sub33	R4 R5 R7
Sub10	R1 R4 R5	Sub22	R2 R4 R7	Sub34	R4 R6 R7
Sub11	R1 R4 R6	Sub23	R2 R5 R6	Sub35	R5 R6 R7
Sub12	R1 R4 R7	Sub24	R2 R5 R7		

simulations. Each subteam had only three homogeneous or heterogeneous robots, but a robot could not be placed into two different subteams at the same time.

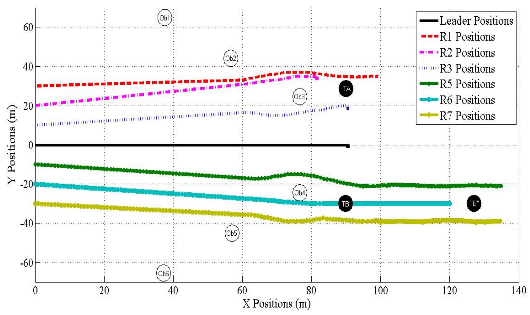
3.1 Simulation - Sim(1)

In Sim (1), the seven robots were holonomic mobile platforms, each of which had two active wheels and did not have a manipulator, and they formed a slave team. The human operator remotely controlled the team leader, R4 while all other slave robots, R1-3 and R5-7, were coordinated with the team leader to approach the targets. In Figure 4a, all robots could avoid the obstacles while keeping a constant distance from each other. With the MRMTMT pairing method in Eqs (1) - (5), the WAGT table for TA and TB, Table 4, was produced based on which the subteams were properly paired to tasks and targets. Bids in Table 4, (Ta, Tb) where Ta was the bid value for TA when Tb was for TB, were calculated in Eq. (5) as an inverse of the sum of target-robot distances in a subteam minus the base price when the base price for t_1 was 30 and t_2 was 10. The reasons were that in order to start with the tasks, the robots needed to maintain at least 30(m) from TB for t_1 when only keeping at least 10(m) from TA for t_2 because the robots need more space to do t_1 than t_2. As results of Table 4, Sub1 was paired to TA for the task of the target capture when Sub35 was paired to TB for the task of the target transportation, and R4 was selected as a team leader.

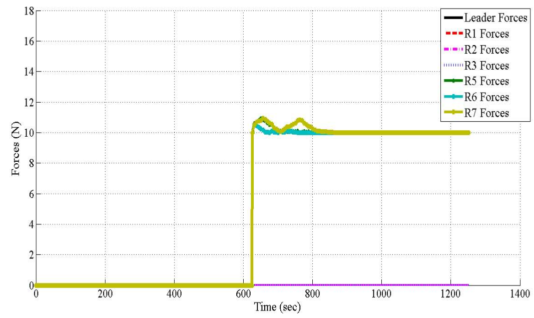
In Figure 4b, the simulation results showed that the contact forces were maintained at the desired value 10 (N) at most of the time. Those recorded contact forces during the simulation were so desired value that a firm grip of TB was achieved while not being too large to break the robots or TB. Moreover, in Figure 5b, the force errors varied from 0.0 to 0.9(N), and a force error average was 0.45 (N) when in Figures 5a, the position errors were recorded from 0 to 0.12 (m), and a position error average was 0.05 (m). The position and force errors were as low as those within an acceptable range when the tasks were completed simultaneously in 1250 seconds. Therefore, the performance of the SMMS teleoperator for MRMTMT pairing was enhanced.

Table 4: Weighted Attack Guidance Table (WAGT) for Target A and B

Subteam	Bids	Subteam	Bids	Subteam	Bids
Sub1	(41,69)	Sub13	(39,73)	Sub25	(38,76)
Sub2	(40,69)	Sub14	(39,74)	Sub26	(39,74)
Sub3	(40,70)	Sub15	(38,75)	Sub27	(39,75)
Sub4	(40,71)	Sub16	(40,71)	Sub28	(39,75)
Sub5	(39,71)	Sub17	(40,71)	Sub29	(39,76)
Sub6	(40,70)	Sub18	(39,73)	Sub30	(38,76)
Sub7	(40,71)	Sub19	(39,73)	Sub31	(38,77)
Sub8	(40,72)	Sub20	(39,73)	Sub32	(38,77)
Sub9	(39,72)	Sub21	(39,74)	Sub33	(38,78)
Sub10	(40,72)	Sub22	(39,74)	Sub34	(38,78)
Sub11	(39,73)	Sub23	(39,75)	Sub35	(38,79)
Sub12	(39,73)	Sub24	(39,75)		

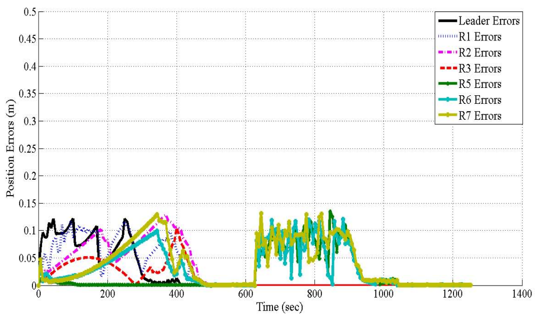


(a) **Sim (1)** - Slave Path Trajectories

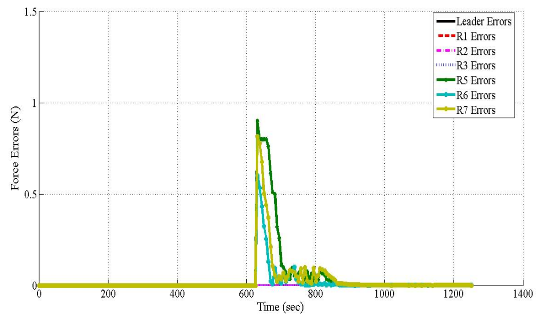


(b) **Sim (1)** - Slave Forces

Figure 4: Simulation - Sim (1) Positions and Forces



(a) **Sim (1)** - Slave Position Errors



(b) **Sim (1)** - Slave Force Errors

Figure 5: Simulation - Sim (1) Position and Force Errors

Table 5: Weighted Attack Guidance Table (WAGT) for Target A and B in Sim (2)

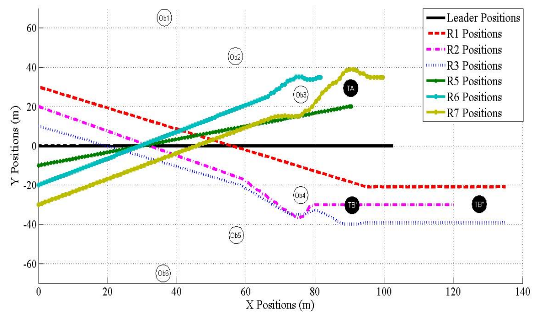
Subteam	Bids	Subteam	Bids	Subteam	Bids
Sub1	(41,369)	Sub13	(239,173)	Sub25	(238,176)
Sub2	(140,269)	Sub14	(239,174)	Sub26	(239,174)
Sub3	(140,270)	Sub15	(238,175)	Sub27	(239,175)
Sub4	(140,271)	Sub16	(240,171)	Sub28	(240,175)
Sub5	(180,271)	Sub17	(240,171)	Sub29	(260,176)
Sub6	(180,250)	Sub18	(239,173)	Sub30	(280,176)
Sub7	(220,225)	Sub19	(239,173)	Sub31	(300,177)
Sub8	(240,200)	Sub20	(239,173)	Sub32	(308,177)
Sub9	(240,190)	Sub21	(239,174)	Sub33	(318,178)
Sub10	(240,172)	Sub22	(239,174)	Sub34	(328,178)
Sub11	(239,173)	Sub23	(239,175)	Sub35	(338,179)
Sub12	(239,173)	Sub24	(239,175)		

3.2 Simulation - Sim(2)

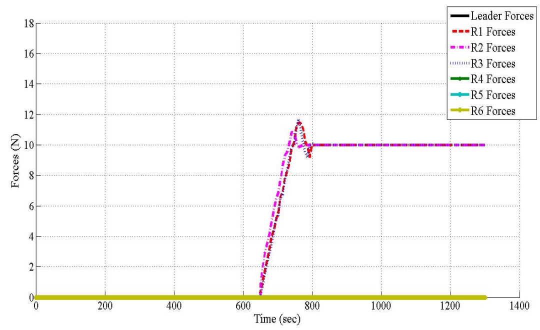
In Sim (2), R1-3 were equipped with a manipulator atop when R4-7 were not. The obstacles, targets, and tasks were equivalent to the ones specified in Sim (1).

With the MRMTMT pairing method, bids in Table 5, (Ta, Tb) where Ta was the bid value for TA when Tb was for TB, were calculated in Eq. (5) as an inverse of the sum of the product of robot capability values and all target-robot distances in a subteam minus the base price when the base price for the transportation was 30 and the capture was 10. In addition, the robot capability values for R1-3 were one for the capture if those were ten for the transportation; however, the robot capability values for R4-7 were ten for the capture if those were one for the transportation. The reasons were that in order to start with the tasks, the robots needed to maintain at least 30(m) from TB for the transportation when only keeping at least 10(m) from TA for the capture because the robots need more space to do transportation than capture. Furthermore, the holonomic mobile platforms with grippers atop were more suitable in a target transportation than the nonholonomic mobile platforms without grippers atop. Hence, every robot had different capability values ranging from zero to one. As results of Table 5, Sub35 was paired to TA for the task of the target capture when Sub1 was paired to TB for the task of the target transportation, and R4 was appointed as a team leader.

In Figure 6b, the simulation results showed that the contact forces were also maintained at the desired value 10 (N) at most of the time, which represented an achievement of a firm grip of TB and no damage of the robot and TB due to too large contact forces. Moreover, in Figure 7b, the force errors varied from 0.0 to 0.85(N), and a force error average was 0.25 (N) when in Figures 7a, the position errors were recorded from 0 to 0.35 (m), and a position error average was 0.08 (m). By comparing the results in Sim(1) and Sim(2), their recorded force and position errors were similar, and their mission completion time was not quite different when the time taken to complete Sim(1) and Sim(2) were 1250 and 1300 seconds, respectively. Therefore, the performance of the proposed SMMS system with the control methods with the MRMTMT pairing method could not be affected if the slave robots are

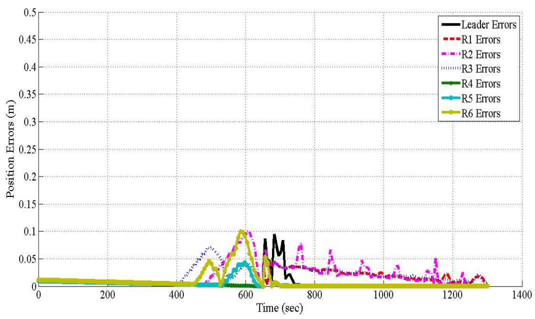


(a) **Sim (2)** - Slave Path Trajectories

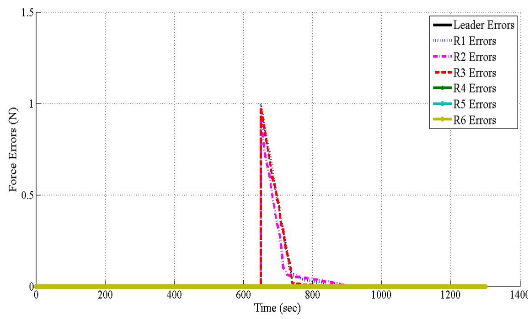


(b) **Sim (2)** - Slave Forces

Figure 6: Simulation - Sim (2) Positions and Forces



(a) **Sim (2)** - Slave Position Errors



(b) **Sim (2)** - Slave Force Errors

Figure 7: Simulation - Sim (2) Position and Force Errors

heterogeneous or homogeneous.

4 Conclusion & future work

The adaptive control method including the above mentioned major components is developed for the SMMS mobile teleoperations to execute multi-tasks on multi-targets and improve the performance in terms of the effectiveness of the task achievement and the system transparency no matter whether the slave robots are homogeneous or heterogeneous as seen from the simulation results. Moreover, heterogeneous robots with the proposed control method can avoid obstacles and track targets because the major components in the method, (1) the modified potential field based leader-follower formation, (2) adaptive master-slave impedance, (3) compensators for contact forces and human errors, and (4) the MRMTMT pairing method, enable each robot to generate its path based on sensed robot-obstacle/target/robot distances and contact forces. Nonetheless, the MRMTMT pairing method could generate a suboptimal solution in general since the MRMTMT pairing algorithm is heuristic.

Therefore, our future work will be to further evaluate the performance of using the MRMTMT pairing method to verify the performance and quality of the pair solutions. In addition, we will look into the proposed control method for a SMMS mobile teleoperator working in much complicated tasks and environments, e.g. an uncertain task that may include unconstrained, constrained, transition, or some motions combining two or all of them in an unknown area, which has not been mentioned in our papers. We will implement the proposed control method into a SMMS system hardware for further experimental validation by comparing the experiments with the simulations to highlight the expected performance enhancements even if the real time delays may vary irregularly.

References

[1] K. R. Baghaei and A. Agah. Task allocation methodologies for multi-robot systems. *Technical Report ITTC-FY2003-TR-20272-01*, 2003.

[2] Z. R. Bogdanowicz and N. P. Coleman. New algorithm for optimization of effect-based weapon-target pairings. *International Conference Scientific Computing CSC 08*, 2008.

[3] Y. Cheung and J. Chung. Cooperative control of a multi-arm system using semi-autonomous telemanipulation and adaptive impedance. *14th IEEE International Conference on Advanced Robotics (ICAR)*, 2009.

[4] Y. Cheung, J. H. Chung, and N. Coleman. Semi-autonomous formation control of a single-master multi-slave teleoperation system. *IEEE Symposium on Computational Intelligence in Control and Automation*, 2009.

[5] Yushing Cheung. *Adaptive semi-autonomous teleoperation of a multi-agent robotic system.* PhD thesis, Stevens Institute of Technology, 2009.

[6] Nak Young Chong, K. Ohba, T. Kotoku, K. Komoriya, N. Matsuhira, and K. Tanie. Coordinated rate control of multiple telerobot systems with time delay. *Systems, Man, and Cybernetics, 1999. IEEE SMC '99 Conference Proceedings. 1999 IEEE International Conference*, 5(12-15):1123–1128, 1999.

[7] M. B. Dias, R. M. Zlot, N. Kalra, and A. Stentz. Market-based multirobot coordination: a survey and analysis. *Proceedings of the IEEE*, 94(7):1257–1270, 2006.

[8] I. Farkhatdinov and Jee-Hwan Ryu. Teleoperation of multi-robot and multi-property systems. *Industrial Informatics, 2008. INDIN 2008. 6th IEEE International Conference*, pages 1453–1458, 1999.

[9] T. Fong, C. Thorpe, and C. Baur. Multi-robot remote driving with collaborative control. *Industrial Electronics, IEEE Transactions*, 50(4):699–704, 2003.

[10] K. Hashtrudi-Zaad and S.E. Salcudean. Adaptive transparent impedance reflecting teleoperation. *Proc. of the 1996 IEEE Int. Conf. Robotics and Automation*, 1996.

[11] Dongjun Lee, O. Martinez-Palafox, and M.W. Spong. Bilateral teleoperation of multiple cooperative robots over delayed communication networks: Theory. *Robotics and Automation, ICRA. Proceedings of the IEEE International Conference*, 2005.

[12] Dongjun Lee and M.W. Spong. Bilateral teleoperation of multiple cooperative robots over delayed communication networks: Application. *Robotics and Automation, ICRA Proceedings of the 2005 IEEE International Conference*, 2005.

[13] Z. J. Lee, C. Y. Lee, and S. F. Su. An immunity-based ant colony optimization algorithm for solving weapontarget assignment problem. *Applied Soft Computing Journal*, 2(1):39–47, 2002.

[14] Bo Liu, Zheng Qin, Rui Wang, You bing Gao, and Li ping Shao. A hybrid heuristic particle swarm optimization for coordinated multi-target assignment. *Industrial Electronics and Applications, ICIEA. 4th IEEE Conference*, pages 1929 – 1934, 2009.

[15] K. Ohba, S. Kawabata, N.Y. Chong, K. Komoriya, T. Matsumaru, N. Matsuhira, K. Takase, and K. Tanie. Remote collaboration through time delay in multiple teleoperation. *Intelligent Robots and Systems, 1999. IROS '99. Proceedings. 1999 IEEE/RSJ International Conference on*, 3:1866–1871, 1999.

[16] J Ota, N. Miyata, T. Arai, E. Yoshida, D. Kurabatashi, and J. Sasaki. Transferring and regrasping a large object by cooperation of multiple mobile robots. *Human Robot Interaction and Cooperative Robots, Proceedings. IEEE/RSJ International Conference*, 1995.

[17] Hyeshin Park, Yo-An Lim, A. Pervez, Beom-Chan Lee, Sang-Goog Lee, and Jeha Ryu. Teleoperation of a multi-purpose robot over the internet using augmented reality. *Control, Automation and Systems, 2007. ICCAS '07. International Conference*, pages 2456–2461, 2007.

[18] J.H. Ryu, D.S. Kwon, , and B. Hannaford. Stable teleoperation with time-domain passivity control. *Robotics and Automation, IEEE Transactions*, 20(2):365–373, 2004.

[19] T. Suzuki, T. Sekine, T. Fujii, H. Asama, and I. Endo. Cooperative formation among multiple mobile robot teleoperation in inspection task. *Decision and Control 2000, Proceedings of the 39th IEEE Conference*, 1(12-15):358–363, 2000.

[20] Z. Wang, M.N. Admadabadi, E. Nakano, and T. Takahashi. A multiple robot system for cooperative object transportation with various requirements on task performing. *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference*, 2(10-15):1226–1233, 1999.

[21] H. Yamaguchi. A distributed motion coordination strategy for multiple nonholonomic mobile robots in cooperative hunting operations. *Decision and Control, 2002, Pro-*

ceedings of the 41st IEEE Conference, 3(10-13):2984–2991, 2002.

[22] Y. Yamamoto and S. Fukuda. Trajectory planning of multiple mobile manipulators with collision avoidance capability. *Proceedings. ICRA '02. IEEE International Conference*, 2002.