2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

WE-Filter: Adaptive Acceptance Criteria for Filter-based Shared Autonomy

Michael Bowman* and Xiaoli Zhang*, *Member, IEEE*

***Abstract*—Filter-based shared control aims to accept and augment an operator's ability to control a robot. Current solutions accept actions based on their direction aligning with the robot's optimal policy. These strategies reject a human's small corrective actions if they conflict with the robot's direction and accept too aggressive actions as long as they are consistent with the robot's direction. Such strategies may cause task failures and the operator's feeling of loss of control. To close the gap, we propose WE-Filter, which has flexible, adaptive criteria allowing the operator's small corrective actions and tempering too aggressive ones. Inspired by classical work-energy impact problems between two dynamic, interactive bodies, both inputs' properties (direction and magnitude) are inherently considered, creating intuitive, adaptive bounds to accept sensible actions. The model identifies behaviors before and after impact. The rationale is that each timestep of shared control acts as an impact between the operator's and the robot's policies, where post-impact behaviors depend on their previous behaviors. As time continues, a series of impacts occur. The aim is to minimize impacts that occur to reach an agreement faster and reduce strong reactionary behaviors. Our model determines flexible acceptance criteria to bound a mismatch of magnitude and finds a replacement action for conflicting policies. The WE-Filter achieves better task performance, the ratio of accepted actions, and action similarity than the existing methods.**

I. Introduction

A. Importance and Needs of Filter-based Shared Control

Filter-based shared control strategies are used for teleoperation scenarios where an operator has a strong grasp of the dynamics, controls, and tasks while desiring more freedom to maneuver the robot in a valid control strategy by not over-constraining their actions [1]. One goal is to have the operator maintain as much control over the robot as possible, which would have more human input actions accepted than rejected. Another goal is maintaining safety, which rejects human input incongruous with optimal robot control signals. These two goals need to be balanced to ensure an operator maintains a sense of control while achieving task success.

The optimal actions of the autonomous agent may differ from the operator in real-world applications. The different actions are inherent to the system for two factors: 1) an operator attempting to handle unexpected uncertainties and 2) multiple ways to accomplish a shared control task. Toward the first factor, operators adjust to overcome the uncertainties in perception and the environment. For example, if the robot misses the initial approach of grasping an object by knocking it over in a teleoperation scenario, an operator will attempt to adjust the actions to correct the robot. However, the robot assumes the pose is correct and rejects these sensible operator actions. In the second factor, the robot's optimal policy in a teleoperation scenario may desire a grasp orientation over the top of the object, while an operator prefers the side to see due to visual constraints. Neither agent has an incorrect action, nor is the robot producing a suboptimal action plan. This difference is fundamentally inherent in shared control, and precautions are necessary to prevent filters from rejecting an operator's sensible actions. Thus, filtered shared control must be flexible toward the operator's corrective actions or preference, empowering them to stay in control.

This work was supported by U.S. NSF under Grants 1652454 and 2114464.(Corresponding author: Xiaoli Zhang.)
*Michael Bowman, and Xiaoli Zhang are with the Colorado School of Mines, Intelligent Robotics and Systems Lab, Golden, CO 80401 USA (email: mibowman@mines.edu, xlzhang@mines.edu).

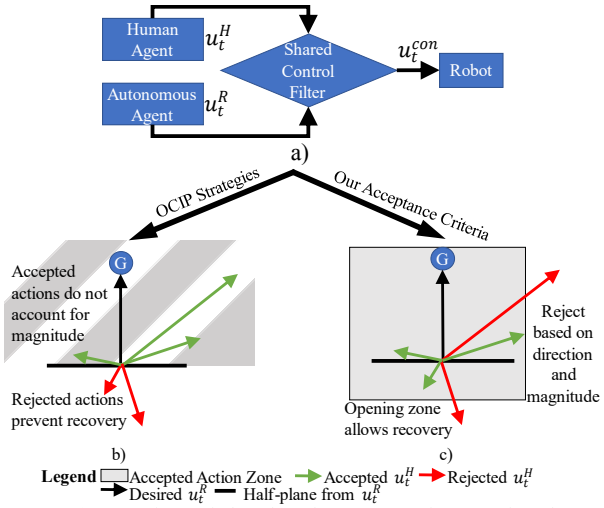

Figure 1: a) Overview of the shared autonomy framework, where our contributions focus on providing an optimization procedure for the Shared Control Filter block as in c). b) is the standard approach rejects actions solely on direction which rejects small corrective actions and accepts strong, aggressive actions towards the goal. c) Our strategy simultaneously looks at direction and magnitude to allow smaller actions by an operator to recover, while tempering strong actions of the operator. The floating, adaptive grey zone in c) represents the zone of acceptable actions.

Current filter-based shared control strategies usually reject actions; there is no option other than conforming to the robot policy. They strictly use a directional filter (Fig. 1b). If the operator's direction differs from the robot policy, the controller will reject the action, and the operator will feel a loss of control. It prevents an operator from 1) resetting to try again, 2) correcting the robot's action, or 3) using it in a preferred way. Fig. 1b shows the result of the direction-only bounds for a single timestep (preventing sensible actions) when an operator tries to back out and try a new way. The consequences are the operator: 1) makes drastic movements to no avail; and 2) must give up on their strategy as they frequently have "risky" actions rejected.

B. Current State-of-the-Art and Open Problems

Original filter strategies such as virtual fixtures aim to prevent operators from damaging a pre-known environment and rely on the operator's action plan as they do not use an optimal robot policy [2][3]. An alternative strategy augments the human actions based on a model-free learned optimal

policy following Deep Reinforcement Learning (DRL) methods [4]. DRL methods assume the operator has an imperfect understanding of the system dynamics and change the operator's actions based on the robot's filter criteria to provide better task performance. Recently, techniques using Maxwell's Demon Algorithm (MDA) provide a model-based structure to determine whether output actions follow the operator or its own optimal policy. The MDA filters take the form of optimal control problems. They use Model Predictive Control (MPC) for nonlinear and discrete systems (rather than DRL). The most recent MDA filter with MPC is known as Sequential Action Control (SAC) [5][6]. It was devised for noise-driven control [7] and has been used for teleoperation [8] by treating operators as noisy rational agents [9][10]. Two open problems exist for filter methods: determining 1) a selection criterion and 2) a post-rejection plan.

Selection criterion: Two state-of-the-art approaches exist for selection criteria. The first is the optimal controller inner product (OCIP), which aims to ensure the inner product is positive, and within an angle tolerance [8]. OCIP calculates a dot product between u_t^R and u_t^H and applies thresholding for acceptable u_t^H [11]. *The underlying assumption behind the OCIP is that a user agrees with the optimal control. This makes OCIP best for teaching an operator the optimal policy as it would reject more inputs that do not conform with the optimal policy* [12]. Another strategy improves OCIP and is called mode insertion gradient (MIG). Unlike OCIP, the MIG does not assume an operator agrees with the optimal control; instead, operators move in a descent direction toward a goal. MIG was initially used for scheduling problems to determine *when* to switch control modes; however, it was adapted to determine *whether* it should switch [12][13]. MIG accepts more inputs than OCIP. However, it still allows strong motions (potentially causing object knock-over) and rejects adjustments common in shared control (Fig. 1b).

Post-rejection plan: The second open problem is determining what to do after rejecting human input [1][8]. There are two main routes: 1) apply no control, or 2) apply the optimal robotic control from MPC (called never-fail solutions). Both have been suggested for different applications and motivations. The first has been used for scenarios requiring higher operator control authority, while the second should be used when an operator is in a supervisory role. The post-rejection plan has no consensus and remains an open problem.

C. Contributions

This work aims to develop a filter shared control strategy called the WE-Filter inspired by classic dynamics work-energy impact problems. The model explicitly identifies behaviors (velocity actions) before and after impact. The rationale for such a model is that each time step of shared control acts as an impact, where post-impact behaviors depend on their previous behaviors. Thus, previous actions are observed while the current robot response is controllable. As time continues, a series of impacts occur. The aim is to minimize impacts that occur to reach an agreement faster and reduce strong reactionary behaviors. This allows corrective actions to be accepted and rejects too-aggressive actions, so an operator does not feel a loss of control. The operators can refine their actions without conforming to the robot policy. We handle two critical problems current filter strategies struggle with: 1) acceptance criteria incompatible with an operator's senses and 2) post-rejection procedures.

When considering direction and magnitude (as in Fig. 1c), we can change bounds in different directions; our new acceptance criteria reflect the realities of the interaction between the operator and autonomous agent. An impact model is developed to account for each agent's magnitude (we call *effort* denoted as W_t) and the relative difference in direction and magnitude (we call *relative disagreement* denoted as e_t). With each new human input, new e_t and W_t^c are evaluated. We aim to maximize W_t which drives the operator and autonomous policies to the known goal. The e_t term acts as a constraint to keep the direction and magnitude within an acceptable mismatch range based on the current and previous actions of the robot and human. Since it is temporal and adaptive, it can restrict the set of actions (grey zone smaller in Fig. 1c than Fig. 1b). It should allow the operator to move backward, albeit in a limited manner. This allows the operator to objectively maintain control of the robot while improving task performance. The contributions follow:

1. Model the interaction between an operator's and autonomous agent's actions using a two-mass dynamic work-energy impact model. It describes current actions to previous ones through effort (magnitude) and relative disagreement (magnitude and direction).
2. Develop adaptive acceptance criteria based on the current autonomous agent and operator's actions through relative disagreement (e_t^R). We accept operator actions outright when an action is admissible ($e_t^R < 1$) and we find a replacement action when it is rejected.
3. Provide an optimization framework when an operator's action is rejected. The controller finds a solution that suits both conflicting policies. We maximize effort (W_t^c) and bound the relative disagreement (e_t^c) to find the action.

II. Methods

A. Current Filter Shared Control Acceptance Criteria

For the remainder of the paper, the current actions of a human and autonomous agent will be denoted as u_t^H and u_t^R, while actions of a previous timestep are u_{t-1}^H and u_{t-1}^R. The applied action to the physical robot will be denoted as u_t^{con} and u_{t-1}^{con}. The OCIP follows a straightforward dot product to determine the direction the control vectors are pointed, shown in Eq. 1:

$$u_t^{con} = \begin{cases} u_t^H, & u_t^H \cdot u_t^R > 0 \\ 0, & u_t^H \cdot u_t^R < 0 \end{cases} \quad (1)$$

Eq. 1 creates a halfplane of acceptable actions, limiting the ability of an operator to back out as the u_t^{con} gives no action. This severely limits accepted actions as dimensions increase. This strategy only looks at the direction of actions, not their magnitude. The OCIP serves as the baseline for evaluation.

B. Shared Control Impact Model

A work-energy impact model describes the dynamic, reactive behaviors of two-point masses when they pull apart

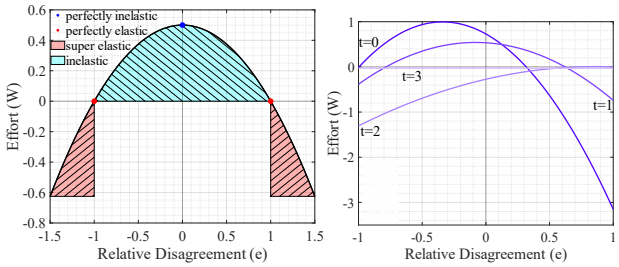

Figure 2: An example curve for actions at a single timepoint (left), and a series of them (right). Four scenarios of interest occur on the left. 1) relative disagreement is perfectly inelastic, this action would be identical to their partner's action and maximum effort was put in to agree (blue point), 2) relative disagreement is inelastic which means we have put in positive effort to agree (blue shadow region), 3) the perfectly elastic collision points mean either no change in actions or a complete swap of actions (red points), 4) super elastic collisions result when an agent is actively trying to put in negative effort towards the team (red shadow regions). On the right, a series of 4 impacts occur. Each impact produces a different shaped curve. At t=0 the shape is similar to the left figure. As time continues, the curve becomes flat at nearly 0 (t=3). Meaning that u_{t-1}^H and u_{t-1}^{con} are the same, and actions reach agreement. The flattening pattern is only true if the team can reach an agreement, otherwise, each time step could stay with sharp convexity.

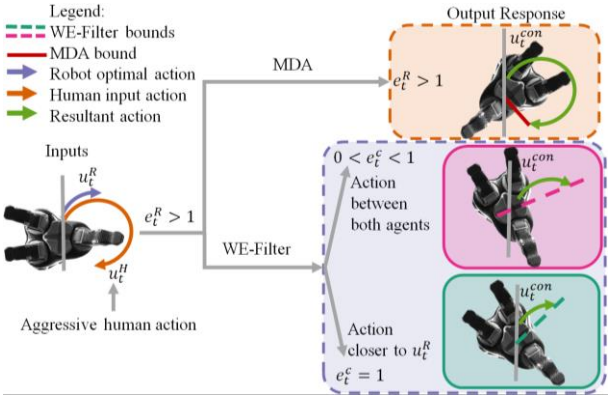

Figure 3: Responses of the MDA and WE-Filter to aggressive human rotation in telemanipulation. MDA strategies allow over rotation (top) when both agents agree on the direction of rotation (e_t^R>1). WE-Filter rejects actions when e_t^R>1 and replaces them by bounding relative disagreement (e_t^c). The e_t^c aids in deciding whether u_t^{con} will have an action more like u_t^R or u_t^H. In this aggressive action scenario, we do not want e_t^c=0 or else it will be identical to MDA output (an action, $u_t^{con} = u_t^H$, we already rejected, e_t^R>1). We limit e_t^c so that u_t^{con} is not so like u_t^R that it does not follow the human.

from one another (u_t^H and u_t^R have different action policies) while adhering to the momentum along a single axis (line of impact). The model explicitly identifies behaviors (velocity actions) before and after impact. The rationale is that shared control acts as an impact at each time step, where new u_t^H and u_t^R behaviors depend on the previous ones (u_{t-1}^H and u_{t-1}^R). The controller is given the four actions and determines whether u_t^H is admissible. As time continues, a series of impacts occur. Two calculations model the interaction.

1) Relative Disagreement: Coefficient of Restitution

The first calculation determines whether u_t^H is admissible or if we need to optimize for an admissible action. It is the coefficient of restitution, e_t, and determines the relative difference in action direction and magnitude. We refer to e_t as *relative disagreement*. e_t^R evaluates u_t^H and u_t^R, while e_t^c evaluates u_t^H and u_t^{con} if optimization is required for the human input. As the equations are identical in form, in this section, we will demonstrate with e_t^R and is defined in Eq. 2.

$$e_t^R = \frac{u_t^H - u_t^R}{u_{t-1}^R - u_{t-1}^H} \quad (2)$$

Four cases of Eq. 2 must be discussed analogously from the dynamics problem to the shared control context. The explanation will be made with Fig. 2 as the mathematical model. Four cases occur: 1) perfectly inelastic collisions, e_t^R =0, 2) inelastic collisions, $0 < |e_t^R| < 1$, 3) elastic collisions, $|e_t^R| = 1$, and 4) superelastic collisions, $|e_t^R| > 1$.

The first case, perfectly inelastic collisions, describes two-point masses stuck together after impact, which mathematically is $u_t^H - u_t^R = 0$. This implies that both agents have a perfect agreement and want the same action to be taken. The second case, inelastic collisions, demonstrates after impact a trend toward full agreement (the blue region in Fig. 2). This occurs when the difference between the current actions is smaller than the previous ones, which means the new actions are becoming more aligned and both agents begin to agree. Mathematically, this implies $|u_t^H - u_t^R| < |u_{t-1}^R - u_{t-1}^H|$. This would be seen as slight differences in poses between u_t^H and u_t^R (refinement). The third case is elastic collisions, where $|e_t^R|$=1 (the two red points which cross the x-axis in Fig. 2). This implies $|u_t^H - u_t^R| = |u_{t-1}^R - u_{t-1}^H|$. This means after impact complete relative disagreement has been achieved. Last is superelastic collisions, $|e_t^R| > 1$ (the two red regions below the x-axis Fig. 2). This case means energy has been introduced into the system, and the actions have become even more polarized, as this implies $|u_t^H - u_t^R| > |u_{t-1}^R - u_{t-1}^H|$. The actions taken have created tension, and an operator overreaction may have occurred.

All cases are important and demonstrate different task behaviors, but none are truly bad. Although case 4 may appear less desirable, it might not be entirely true without context for the change of actions. For instance, consider u_t^H grasping an object's side while u_t^R grasps the top. The human may make a sudden change in their preferred orientation, which induces case 4 because they want to see the grasp configuration easier. Although, designers will likely aim for case 1 and case 2, as this implies the operator follows the optimal robot policy. In II.C, we describe how our controller would find a replacement action if case 4 occurs as u_t^H would be deemed too large to be a corrective one. Our criteria for an admissible u_t^H is if e_t^R is in case 2, $|e_t^R| < 1$; otherwise, an optimization process is needed to determine a better u_t^{con} so that $|e_t^c| \leq 1$. In Fig. 3, we showcase 4 ($|e_t^R| > 1$) occurs if we use the MDA controller. Our method will adjust the u_t^{con} to either be in case 2 (0<e_t^c< 1, in the middle row of Fig. 3), or to case 3 (e_t^c=1, in the bottom row of Fig. 3).

Another critical component is when e_t^R<0. This implies that one agent has stayed more dominant for both time steps, such as $u_t^R > u_t^H$ and $u_{t-1}^R > u_{t-1}^H$ or $u_t^R < u_t^H$ and $u_{t-1}^R < u_{t-1}^H$. Lastly, an immediate reaction to$u_{t-1}^R - u_{t-1}^H = 0$ may jump out; however, this is acceptable. When the previous actions align, new actions appear as splitting hairs over fine details. This occurs when the $u_{t-1}^R - u_{t-1}^H$is very small compared to the $u_t^R - u_t^H$, which accounts for and demonstrates the recency bias [16] [17]. This can be limited within our optimization by either applying bounds as written above or adding numerical tolerances to Eq. 2's denominator.

2) Effort: Residual Energy Calculation

The second calculation is the residual energy in the system and is used to guide the optimization. The second calculation has two forms like e_t, W_t^R and W_t^c.

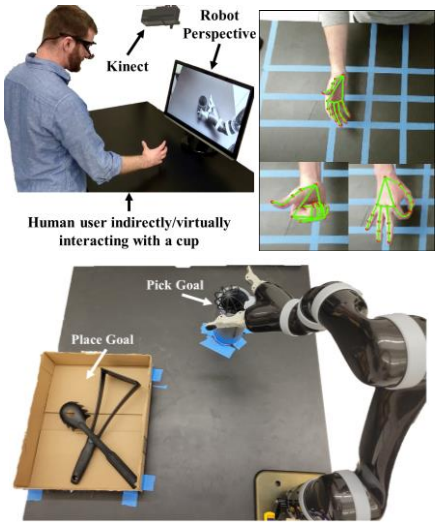

Figure 4: Operators see the robot perspective on the screen while they move their hand freely over the table. The Xbox Kinect captures hand posture to extract the palm pose and the amount each finger is open or closed. The information is then sent to the robot to execute the motion.

$$W_t^c = \frac{m_R}{2}((u_{t-1}^{con})^2 - (u_t^{con})^2) + \frac{m_H}{2}((u_{t-1}^H)^2 - (u_t^H)^2) \quad (3)$$

Here, we refer to W_t^c as the *effort* both agents put in towards the interaction as it quantifies the actions' magnitude changes. Ultimately, the goal is to have W_T^c converge to 0 (where t = T is the terminal time horizon) as they progress in the task, implying both agents have settled on a velocity action (seen in the right of Fig. 2 with the faint t=3 line). Toward this high-level goal, it is achieved by maximizing W_t^c at each time step, which would be equivalent to minimizing u_t^H(which we cannot control), and u_t^{con}. Since these actions are in terms of velocities, it would be ideal if u_t^{con} and u_t^H are driven to zero over time, which implies they settled on a state. The parameters m_R and m_H are the weighting terms which can act as role allocation. In a peer-peer setting, they are set to 0.5, while a leader-follower scheme may differ to values such as 0.9 and 0.1. More details on roles can be found in [14] and [15]. In this work, both terms are set to 0.5.

With Eqs. 2 and 3 established, interesting relationships can be seen if we consider the system controllable through the lens of the u_t^{con} action as a u_{t-1}^{con}, u_{t-1}^H, and u_t^H are observed. Formally, we can look at the relationship of W_t^c and e_t^c by combining Eqs. 2 and 3 through u_t^{con} in Eq. 4 and Fig. 2.

$$W_t^c = -\frac{1}{2}m_R\,(u_{t-1}^H - u_{t-1}^{con})^2 e_t^{c\,2} - m_R u_t^H (u_{t-1}^H - u_{t-1}^{con}) e_t^c + \frac{1}{2}m_R (u_{t-1}^{con})^2 + \frac{1}{2}m_H (u_{t-1}^H)^2 - \frac{1}{2}(m_R + m_H)(u_t^H)^2 \quad (4)$$

With Eq. 4, u_t^{con} impacts variable e_t^c directly as all other actions are given. The quadratic relationship of e_t^c and W_t^c is helpful as we can achieve a global maximum (seen in Fig. 2).

The parabolic shape means the behavior can shift left, right, up, or down, and the convexity changes too. This means that the maximum of W_t^c is not always at e_t^c=0. The maximum W_t^c could reside in $|e_t^c|$>1. It is common that current u_t^H have greater magnitude than the previous u_{t-1}^H due to corrective refinement motions meaning the maximum will be $W_t^c < 0$ or $|e_t^c|$>1. Therefore, within acceptable bounds (-1 < e_t^c < 1), the aim is to obtain the maximum W_t^c. Due to this possibility, many interactions could result in $|e_t^c|$=1.

C. We-Filter: Acceptance Criteria and Optimization

The optimization is designed for the telemanipulation scenario; an operator has an independent one-to-one mapping to the robot. u_t^H, u_t^R, and u_t^{con} are $\mathbb{R}^7$, where the first 6 DOFs are the end-effector pose velocities (position and orientation velocities), and the last DOF controls the velocity to open and close the hand. So, an operator can assume direct control over the robot's end-effector pose. For each component, we check if it is within the e_t^R bounds. If it is, we accept it and move on to the next component. If it is not, a suitable solution is found. This process occurs for each new human input received.

$$u_t^{con} = \begin{cases} u_t^H, & |e_t^R| < 1 \\ Eq.6, & |e_t^R| \geq 1 \end{cases} \quad (5)$$

The optimization aims to determine a new u_t^{con} that is within an e_t^c range (thus relatively similar to u_t^H) while maximizing W_t^c. Fig. 2 shows W_t^c is a convex function in terms of e_t^c. Evaluating each DOF creates a box around possible solutions.

$$u_t^{con} = argmax\, W_t^c \quad s.t. \quad -1 \leq e_t^c \leq 1 \quad u_t^{con} \in U \quad (6)$$

A parallel optimization is employed for each dimension of u_t^{con} where U are the environmental bounds and physical limitations of the robot. In our experimental scenario, the bounds were set such that u_t^{con}'s operation range was between u_t^H and u_t^R. The initial value was set to u_t^R. At each time step, the optimization observes a new u_t^H and the previous command, u_{t-1}^H, are given. The u_{t-1}^{con} is the previous command sent to the robot. This makes u_t^{con} the only variable to be optimized is thus the controller's output to run the physical robot. The rationale for this optimization strategy is two-fold: the output action will be 1) an action that is between both u_t^R and u_t^H within e_t^c bounds while 2) the bounded convexity can ensure an optimal action. This provides an effective bound to allow an operator to make corrective actions (refinement of pose near a goal) and tempers their aggressive erroneous ones (sudden jumps in a posture like our example of $|e_t^R|$>1 when u_t^H decides to grasp the object on the side rather than the top). This ensures the robot does not jump ahead of the operator (potentially knocking the object over) or stall out, resulting in them losing control.

III. Experimental Results

A. Telemanipulation Setup

The telemanipulation experimental setup is described in Fig. 4. An RGB-D image is taken with an Xbox Kinect, and features are extracted using MediaPipe [18] before being used as inputs for the robot control. The operator controls the hand pose and the amount each finger opens and closes. In all, u_t^R is $\mathbb{R}^7$. The operator aims to grasp a regular kitchen utensil from a holder and place it in a bin. This aims to recreate an open pick and place task so the operator can move however they wish to complete the task. The operator can complete tasks with different strategies and recovery plans. With no time constraints, operators have multiple attempts to grasp and place the object in the bin. The robot is given a noisy goal pose (within ±2cm) for both the pick and place locations.

Operators were given unlimited training time to familiarize themselves with both control strategies to mitigate bias.

The institution's Institutional Review Board (IRB) approved a set of experiments. Before participating in the study, written consent was obtained, acknowledging they understood the robot setup, the purpose of the study, and the potential risks involved (arm soreness). Three trials were conducted for each filtered strategy by each participant. A trial consisted of three runs; each run was a different object to grasp. A randomized order of the control modes was used to reduce learning effects. As a pilot study, 10 participants volunteered (a total of 90 runs for each mode). The volunteers were 24.1 ± 3.14 years old and included five women and five men with no prior experience controlling the robot. Due to being a pilot study for the viability of the control paradigm, the data analysis had unique scrutiny for the smaller sample size.

B. Telemanipulation Results

Four quantitative criteria are used to evaluate the effectiveness of the filter shared control strategies: 1) the success rate, 2) completion time, 3) cosine similarity, and 4) the ratio of accepted actions. Table I shows the summary statistics for the measures. Due to the smaller number of trials, necessary precautions must be considered for each measure. The success rate uses a Laplace Estimate rather than the Maximum Likelihood Estimate to reflect the true success rate. An adjusted-Wald 95% confidence interval was used. An N-1 Chi-squared test was conducted to determine statistical significance. The WE-Filter (64%) outperforms the MDA filter (39%) with statistical significance ($p < 0.001$). The operators had a better success rate in completing the task. This is likely due to the operator's actions being accepted near the goal as they have different desired poses. The operator can handle the uncertainty, and the robot affords them the chance to complete the task in their way.

The analysis for the time data is not normally distributed but rather positively skewed [19]. Therefore, a log transform is necessary. The geometric mean and a 95% confidence interval are used. A two-sample t-test is used to determine the statistical significance. The time data are only the successful completion time trials. The WE-Filter (38.60 seconds) outperforms the MDA filter (56.23 seconds) with statistical significance ($p < 0.05$). The faster time data shows operators are not constantly being rejected by the filter and can functionally make actions to refine the robot pose to complete the task. Both task performance metrics show improvements over the MDA filter. However, the two other measures can reinforce the notion of an operator having better control over fine-tuning the robot pose to complete the task.

TABLE I: TELEOPERATION RESULTS FOR THEIR RESPECTIVE MEANS AND CONFIDENCE INTERVALS

Mode	Success rate	Completion time (s)	Cosine distance	Ratio of accepted actions
WE	**0.64 [0.54,0.74]**	**38.60 [32.53,45.80]**	**0.206 [0.187,0.228]**	**0.973 [0.971,0.974]**
MDA	0.39 [0.30,0.49]	56.23 [43.97,71.90]	0.589 [0.560,0.619]	0.681 [0.678,0.684]

WE means WE-Filter, MDA means Maxwell's Demon Algorithm. Bold means this mode performs better in this metric by statistical significance.

The other two measures of cosine distances and the ratio of accepted actions are described in Eqs. 7 and 8, respectively, with u^H and u^{con}. The cosine distance measure is:

$$\sum_t 1 - \frac{u_t^H \cdot u_t^{con}}{\|u_t^H\| \|u_t^{con}\|} \tag{7}$$

The aim is to determine the alignment of the control vectors for the controller output and the operator action. A score of 0 is considered optimal (they perfectly align). The geometric mean and 95% confidence intervals were generated and placed in Table I. A two-sample t-test was used to determine the statistical significance. The WE-Filter outperforms the MDA filter ($p < 0.001$) as the output controller's actions are more similar to human actions. This is because of the filter design, where our approach aims to align actions more closely with an operator rather than outright reject them. This is more evident when analyzing the ratio of accepted operator actions:

$$\phi = \frac{\sum ag}{number\ of\ actions},\ 0 \le \phi < 1,\ \ ag = \begin{cases} 1, if\ u^H \cdot u^{con} > 0 \\ 0, if\ u^H \cdot u^{con} \le 0 \end{cases} \tag{8}$$

This metric determines the number of operator actions accepted by the controller. The aim is to normalize the number of actions taken and determine a better ratio of agreeable actions. A score of 1 is considered optimal as this would imply that all user actions were accepted. The geometric mean and 95% confidence intervals were generated and placed in Table I. The two-sample t-test is used to determine the statistical significance. The WE-Filter outperforms the MDA filter ($p<0.001$) as the ratio of accepted actions is greater. This is expected as the MDA filter rejects u_t^H rather than augments the parts of u_t^{con} to align with the operator. Thus, it is sensible to conclude that our filter benefits the operator in their preferred strategy. To further assess whether the assistance benefits the operator, we can look at u_t^H and u_t^{con} for qualitative analysis.

Qualitative analysis can be shown in two ways: 1) how human input relates to robot output response and 2) how an MDA output compares to the WE-Filter output. Fig. 5 shows

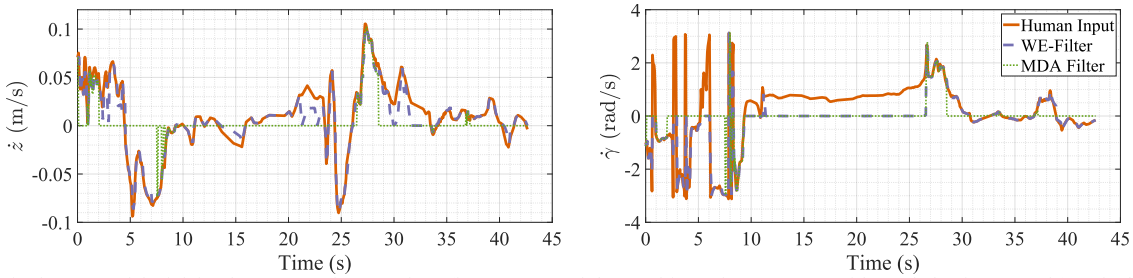

Figure 5: A single successful trial for the operator input actions (in orange), and the resulting robot response with our behaviors (purple) and what would be the MDA response(green). Only 2 dimensions (actions in height and yaw) are shown for easier viewing. If our approach deems the actions too aggressive, the output action will restrict the motion (e.g., in $\dot{z}$ during ~20s to ~23s). When the MDA filter rejects an action, the action is set to 0. Our approach accepts more actions on the whole (e.g., in $\dot{\gamma}$ from ~26s to 45s), yet rejects too aggressive motions (e.g., in $\dot{\gamma}$ during the first 8s) and replaces them with acceptable actions.

a successful trial for the task. Although actions are opened up using this adaptive criterion in some areas, it also restricts the operator's actions, as seen in the $\dot{\gamma}$ where both actions are nearly similar at 0. Our approach accepts more actions than the MDA filter. Notice large blocks where the MDA filter would reject the operator inputs flat out for $\dot{z}$ and $\dot{\gamma}$, while our approach will allow refinement in the $\dot{z}$ (from ~11s to~26s). This is seen in the $\dot{\gamma}$ from 0s to ~8s. The WE-Filter rejects sudden reactions while accepting ones more in line with previous actions. Strong reactions will knock down the object if they are not limited. The WE-Filter accepts more human actions and allows greater refinement over the MDA strategy.

Fig. 6 shows the W_t^c and e_t^c plots for the left side of Fig. 5 (the $\dot{z}$). Our approach drives W_t^c to 0 over time to a goal. The oscillations at the beginning of the trajectory and in the middle indicate that the operator is moving towards the pick goal and place goal, respectively. The oscillations demonstrate when an operator is unsure of their actions. Our filter aims to settle them by finding the maximum W_t^c within the e_t^c bounds at each timestep. By doing so, the W_t^c is driven to 0 over time. The periods when e_t^c are 0 shows when u_t^{con} deems the most effective solution is the u_t^H (i.e., alignment of u_t^H and u_t^R). The timepoints where e_t^c =|1| demonstrate the u_t^{con} aligns with u_t^R more than u_t^H because the best action to take is closest to the u_t^R. e_t^c does not prevent faster speeds; rather, it rejects sudden speed changes. When W_t^c is very small and smooth, this is seen as the output action, and the team is splitting hairs over minor details. In conclusion, our filter works as expected and provides an effective solution to drive high-effort actions within acceptable agreement bounds. A supplemental video shows examples of both strategies.

IV. Discussion

Two cases of operator behavior need to be discussed: 1) large decisive actions and 2) small refinement actions. The MDA filter allowed the operator to make larger motions than the WE-Filter. This sometimes allowed an operator to approach the object quicker, but it would also cause them to knock the object over more often (shown by a lower success rate). The WE-Filter reduces the magnitude of the actions, so it may not allow large actions like the MDA filter, which would theoretically limit the speed of completing tasks. However, in practice, WE-Filter tempers the operator's aggressive actions resulting in higher success rates and better completion times. Refinement actions are when the WE-Filter truly shined, as evident by the higher ratio of accepted actions and higher similarity for control vectors. Qualitatively, this is best seen in the last recovery case in the video, where an operator adjusts the object pose to prevent collisions. It also allows operators to adjust the rotation of the end-effector while making small adjustments in position to line up the grasp. The MDA struggles with small refinement actions needed to succeed in telemanipulation as more actions would be rejected near the final goal states resulting in the robot stalling out and rejecting operator motion. These refinements by the operator simply exist (i.e., the operator wants a pose they can view on the camera to line up a better grasp or think a certain pose will yield a more stable grasp). These refinement actions are critical in telemanipulation [20].

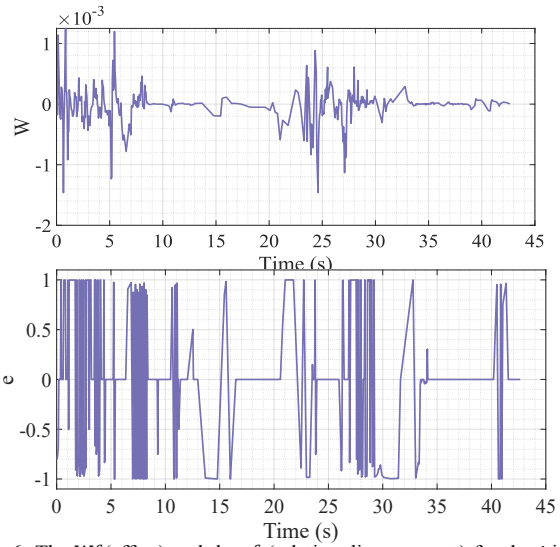

Figure 6: The W_t^c(effort) and the e_t^c (relative disagreement) for the $\dot{z}$ in Fig. 5. Oscillations in the beginning occur as the operator is unsure of their actions. Our approach continues to maximize W_t^c at each timestep in the e_t^c bounds. Ultimately, our solution reduces W_t^c to 0. This holds when a goal is changed (the second oscillation happens when the operator moves to the place goal at ~23s). When actions appear similar in Fig. 5, if e_t^c=0, our strategy deems the best action is the operator's, while an e_t^c=1 or e_t^c=-1 is seen as splitting hairs.

Although the WE-Filter does benefit the operator by adjusting output actions, there are still shortcomings with the current strategies. Potential limitations of our approach are determining 1) hyperparameters for m_R and m_H, 2) the best temporal points to choose (i.e., comparing u_t^H to u_0^H over u_{t-1}^H), and 3) the sensitivity towards noisy inputs by an operator. Investigating these limitations further will allow a model predictive control such as SAC with appropriate roles to shine. The role allocation is outside this work's scope but influences how much the robot will limit its actions over the operator. The second is determining when to consider the impact model, especially if a SAC strategy is adopted. The strategy adopted in this work was using every new operator input (i.e., u_t^H and u_{t-1}^H), but a further investigation must be done to consider more viable time points. For instance, rather than the previous action, like our strategy, it may be the last intervention action applied by the SAC strategy. The third limitation is that noisy human signals occurring when the operator holds their hand steady could cause a vibrating effect on the robot. Although the optimization would attempt to reduce this behavior, a stochastic implementation needs further inquiry. Bounds on e should be task-dependent. Although we follow a classical dynamics threshold, it is not unreasonable to say the selection should be either $|e| \leq 0.5$ for a narrower set of actions, or $|e| \geq 1$ as an adversarial agent to act against the operator for rehabilitation purposes.
Overall, the strategy presented in this work has viability for telemanipulation scenarios. The limitations and proposed future directions demonstrate there is room to improve the model before being used for an MPC-style strategy like [21]. However, the WE-Filter should be considered for future strategies where two agents are attempting to assume control, and a resultant intuitive action is desired.

REFERENCES

[1] A. Broad, T. Murphey and B. Argall, "Operation and Imitation Under Safety-Aware Shared Control," in *Algorithmic Foundations of Robotics XIII: Proceedings of the 13th Workshop on the Algorithmic Foundations of Robotics* 13, pp. 905-920, 2020.

[2] J. Webb, et. al, "Using visuomotor tendencies to increase control performance in teleoperation," in *American Control Conference (ACC)*, Boston, 2016.

[3] F. Ryden, H. J. Chizeck, S. N. Kosari, H. King and B. Hannaford, "Using kinect and a haptic interface for implementation of real-time virtual fixtures," in *Robotics: Sciences and Systems*, Los Angeles, 2011.

[4] S. Reddy, A. Dragan and S. Levine, "Shared Autonomy via Deep Reinforcement Learning," in *Robotics: Science and Systems*, Pittsburgh, 2018.

[5] A. R. Ansari and T. D. Murphey, "Sequential Action Control: Closed-Form Optimal Control for Nonlinear and Nonsmooth Systems," *IEEE Transactions on Robotics,* vol. 32, no. 5, 2016.

[6] I. Abraham, G. De La Torre and T. D. Murphey, "Model-Based Control Using Koopman Operators," in *Robotics: Science and Systems*, Cambridge, 2017.

[7] E. Tzorakoleftherakis and T. D. Murphey, "Controllers as Filters: Noise-Driven Swing-Up Control Based on Maxwell's Demon," in *IEEE Conference on Decision and Control*, Osaka, 2015.

[8] K. Fitzsimons, E. Tzorakoleftherakis and T. D. Murphey, "Optimal Human-In-The-Loop Interfaces Based on Maxwell's Demon," in *American Control Conference (ACC)*, Boston, 2016.

[9] M. Kwon, et. al, "When Humans Aren't Optimal: Robots that Collaborate with Risk-Aware Humans," in *International Conference on Human-Robot Interaction (HRI)*, Cambridge, 2020.

[10] D. Fridovich-Keil et al., "Confidence-aware Motion Prediction For Real-Time Collision Avoidance," *International Journal of Robotics Research,* 39(2-3), pp.250-265., 2019.

[11] A.Broad, I. Abraham, T. Murphey and B. Argall, "Data-driven Koopman Operators for Model-Based Shared Control of Human-Machine Systems," *International Journal of Robotics Research*, 2020.

[12] K. Fitzsimons, A. Kalinowska, J. P. Dewald and T. D. Murphey, "Task-based Hybrid Shared Control for Training Through Forceful Interaction *International Journal of Robotics Research*, 39(9), pp.1178-1195, 2020.

[13] A. Kalinowska, K. Fitzsimons, J. Dewald and T. D. Murphey, "Online User Assessment for Minimal Intervention During Task-Based Robotic Assistance," in *Robotics: Science and Systems*, Pittsburgh, 2018.

[14] A. Mortl, M. Lawitzky, A. Kucukyilmaz, M. Sezgin, C. Basdogan and S. Hirche, "The role of roles: Physical cooperation between humans and robots," *International Journal of Robotics Research* 31, no. 13: 1656-1674*,* 2012.

[15] D. P. Losey et .al, "A review of intent detection, arbitration, and communication aspects of shared control for physical human-robot interaction,"*Applied Mechanics Reviews*, 70(1) 2018.

[16] P. Kaniarasu, et. al, "Robot Confidence and Trust Alignment,"in *International Conference on Human-Robot Interaction (HRI)*, Tokyo, 2013.

[17] M. Desai, P. Kaniarasu, M. Medvedev, A. Steinfeld and H. Yanco, "Impact of Robot failures and feedback on real-time trust," in *International Conference on Human-Robot Interaction (HRI)*, Tokyo, 2013.

[18] C. Lugaresi et al., "Mediapipe: A framework for perceiving and processing reality," in *Third Workshop on Computer Vision for AR/VR at IEEE Computer Vision and Pattern Recognition (CVPR)*, Long Beach, 2019.

[19] J. Sauro and J. R. Lewis, "Average task times in usability tests: what to report?," in *Proceedings of the SIGCHI conference on human factors in computing systems.*,Atlanta, 2010.

[20] M. Bowman, S. Li and X. Zhang, "Intent-Uncertainty-Aware Grasp Planning for Robust Robot Assistance in Telemanipulation," in *IEEE International Conference on Robotics and Automation (ICRA)*, Montreal, 2019.

[21] A. Broad, et. al, "Highly parallelized data-driven MPC for minimal intervention shared control," in *Robotics: Science and Systems*, Freiburg im Breisgau, 2019.