Force Field-Based Indirect Manipulation Of UAV Flight Trajectories*

Werner Alexander Isop[1] and Friedrich Fraundorfer[1]

***Abstract*— For a variety of applications remote navigation of an unmanned aerial vehicle (UAV) along a flight trajectory is an essential task. For instance, during search and rescue missions in outdoor scenes, an important goal is to ensure safe navigation. Assessed by the remote operator, this could mean avoiding collisions with obstacles, but moreover avoiding hazardous flight areas. State of the art approaches enable navigation along trajectories, but do not allow for indirect manipulation during motion. In addition, they suggest to use egocentric views which could limit understanding of the remote scene. With this work we introduce a novel indirect manipulation method, based on gravitational law, to recover safe navigation in the presence of hazardous flight areas. The indirect character of our method supports manipulation at far distances where common direct manipulation methods typically fail. We combine it with an immersive exocentric view to improve understanding of the scene. We designed three flavors of our method and compared them during a user study in a simulated scene. While with this method we present a first step towards a more extensive navigation interface, as future work we plan experiments in dynamic real-world scenes.**

I. INTRODUCTION

Navigation tasks in remote scenes are of great importance. Besides of gaming or computer aided design (CAD), applications also include navigation along flight trajectories of an unmanned aerial vehicle (UAV) during search and rescue (SAR) missions. One major goal is to prevent the UAV from flying through impassable or hazardous areas. Besides of changes of the scenes geometry, also other sudden hazardous situations might occur, like sparks from broken power lines, gas leaks or similar. Such could be hard to detect at a distance with the onboard sensors of the UAV, but rather recognized and assessed by the intuition of the remote operator. In this respect, understanding of the scene is vital during navigation to improve situation awareness (SA). Importantly, free-flight navigation from non-immersive egocentric views is most often very difficult. According to Chen et al. [1] a narrow field of view (FOV), orientation and altitude misjudgment could lead to general lack of understanding the scene. Thus, remote operators might by mistake access impassable or hazardous areas, provoking damage of the UAV. If SAR applications are concerned, this emphasizes the necessity of methods to achieve best SA and avoid dangerous situations, ultimately enabling efficient navigation.

One solution is to let the remote operator navigate in an immersive environment. Such environments are able to provide an increased FOV [2] on either purely virtual or physical remote scenes. As a result, lack of SA and the danger of navigating hazardous areas can be significantly decreased. Other approaches limit the motion of the remote operator by spatially constraining the motion trajectory and reducing the interactions to a minimum at the same time [3]. The overall goal is to avoid navigating impassable areas of the scene and to make the navigation more efficient.

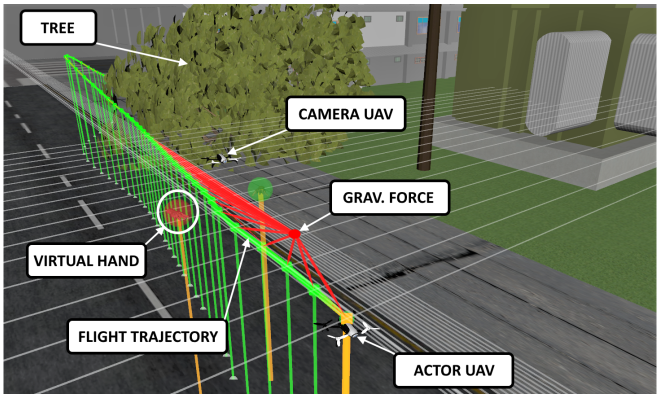


Fig. 1: A remote operator observes a disaster scene from a CAMERA UAV (exocentric view) and indirectly hand-manipulated the flight trajectory of the ACTOR UAV with the MIRROR method flavor to avoid a collapsed tree.

Once a scene is explored, UAV flight trajectories are most commonly defined by global feasible paths. However, keeping track of the sudden presence of hazardous areas can be a difficult and time consuming task. Hazardous areas could be collapsing geometry, but moreover torn power lines or damaged structures. In a simple case, they are difficult to detect by the UAVs sensory equipment because of they are out of range. Hence, the UAV is in danger to pass hazardous areas, potentially getting damaged. To cope with this issue, our work suggests to improve the SA of the operator, whereas improving understanding of the scene enables efficient interaction with the flight trajectory. We focus on cases when a UAV is moving along a flight trajectory that becomes hazardous to navigate, due to sudden environmental changes. In particular, we make the following contributions.

First, we introduce a novel indirect manipulation method, based on *Newton's law of universal gravitation*, and suggest to use it for manipulation of UAV flight trajectories from an immersive exocentric view (Fig. 1). Compared to direct manipulation methods, which are commonly used in egocentric virtual views (Pick-and-Place, GoGo or Homer [4]), our method supports interaction at far distances due to its indirect characteristics. In particular, we aim for conserving smoothness of the flight trajectory after manipulation, whereas our

*This work was supported by the FWF grant I1681.

[1] Werner Alexander Isop and Friedrich Fraundorfer are with the Institute of Computer Graphics and Vision, Department of Computer Science, Graz University of Technology, 8010 Graz, Austria. mail:{isop,fraundorfer}@icg.tugraz.at

method does not require any additional constraints or models (e.g., spring-damper) between the interpolation points. Further, to our understanding we are the fist who even evaluate indirect UAV trajectory manipulation during flight.

Second, we add an additional exocentric viewpoint to the common egocentric view of the flight trajectory. This is enabled by a second companion UAV which follows the main UAV while moving along the trajectory. The exocentric viewpoint is adaptive and the goal is to support the usage of our proposed manipulation method and to improve SA.

We designed three flavors of our indirect manipulation method and compared them against each other in a user study. We present results and extensive discussion of the best performing methods.

II. RELATED WORK

A rich variety of existing work addresses remote free-flight navigation from a single egocentric virtual viewpoint. For example, during remote operation within the use of an immersive head mounted display (HMD), Higuchi et al. [5] map movements of the operator's head directly to the movements of a small sized UAV. In contrast, Mirk and Hlavacs et al. [6] introduce a virtual tourist application, where they do not give the operator full control over the UAV and thus constrain its motion. To increase easiness of use, they map movements of only one rotational axis of the operator's head to the UAV. Their goal is to decrease danger of collisions. Other works, like Hansen et al. [7], do not allow full control of the UAV giving direct motion commands, but introduce a gaze-based interface. They also aim to improve intuitiveness of control, while problems with potential collisions, lack of orientation and loss of SA in general remain.

Other flight interfaces address the problem of collision-free navigation by introducing exocentric views on the UAV and the surrounding scene. For example, Thomasson et al. [8] enable a remote operator to navigate a UAV inside complex 3D structures. They combine a virtual exocentric view onto the UAV and further add an adaptive view method to improve SA and to avoid collisions. However, their work mainly focuses on narrow indoor environments and does not include navigation along global feasible motion trajectories in wide area scenes. In addition, their exocentric viewpoint is designed to be purely virtual. Using live reconstructed 3D data, it is generated in close range of the observed UAV.

Besides of free flight control, Balakrishnan et al. [9] and Burtnyk et al. [10] let the operator interactively steer a camera along a purely virtual, predefined path, whereas they also aim to combine the collision free motion with more simple, intuitive interactions. Moreover, Mirhosseini et al. [3], also suggest navigation along a constrained path. While they discuss an intuitive method for adaptive speed and direction control, however, their path is also predefined in a static scene and lets operators move in egocentric views only. In addition, they do not enable the operator to interactively manipulate the flight trajectory.

Direct and indirect manipulation methods in immersive environments are comprehensively surveyed by Mendes [11], whereas more recent work of Mendes et al. [12] suggests a novel approach for out-of-reach manipulation at far distances. Mendes clearly emphasizes that interaction with objects out of the user's arms reach poses significant challenges and, moreover, direct methods fail. Remarkably, none of these works address our indirect manipulation method based on a gravitational force field.

Manipulation of trajectories, but based on constraints, is well understood. For example, Fowler et al. [13] extensively discuss manipulation of a given curve in either 2D or 3D, also discussing interactive methods to adjust its shape. Furthermore, Michalik et al. [14] suggest to use B-spline shaped curves and surfaces for computer aided design applications. Remarkably, Masone et al. [15] present a shared control framework for online trajectory generation of mobile robots. While they evaluate different interactive methods for direct path manipulation with haptic feedback, they suggest to online retain feasibility during UAV navigation [16]. However, these works discuss manipulation of a given curve based on extensive geometry-constraint models. Further, they do not consider indirect manipulation during a navigational task and do not investigate on a virtual influence on the curves shape which is based on Newton‘s Law of universal gravitation.

To the best of our knowledge, the closest related work to ours is introduced by Gao et al. [17]. They discuss interactive planning and manipulation of feasible paths in form of nanotubes. Their overall goal is to provide an intuitive design-interface for optimal and collision-free planning of electrical connections in nano electromechanical systems. By using a virtual probe, they let the operator manipulate the shape of the tubes from an exocentric viewpoint. Additionally, they model force interactions between the probe and the environment to avoid collisions. However, their approach involves direct manipulation only. Further, they model interdependence between all virtual interpolation points of the tube. Finally, they introduce an exocentric view on the scene, but do not discuss manipulation of the tube during motion.

In contrast to related work, we consider navigation along UAV flight trajectories. We assume a sudden changing environment and focus on use cases where efficient local manipulation of the trajectory is vital, due to unforeseen changes of the situation. To this purpose, we introduce a novel indirect manipulation method and combine it with an exocentric viewpoint. Thus, we let a remote operator safely navigate the UAV in an intuitive and efficient way. We evaluate our method with an extensive simulation framework during a user study and present effectiveness during a virtual SAR mission[1].

III. METHOD

We present a novel method for indirect manipulation of UAV flight trajectories, whereas with this work we reveal two aspects. First, we suggest a physical law which even

[1] Supplementary video: https://youtu.be/f43RCJWs7Is

allows us to efficiently model the interaction between the remote operator and the trajectory. We combine this law with well established methods in virtual immersive environments to enable intuitive manipulation, based on the operator's hand movements. Second, we introduce a virtual exocentric viewpoint, improving SA. While during navigation and manipulation of the trajectory optimal views are important, this work in the first step is mainly focused on the indirect manipulation method.

A. Hand Interaction

Our indirect manipulation method is based on hand interaction for virtual environments, whereas the principle relates to the GoGo technique. Introduced by Poupyrev et al. [18], GoGo suggests to combine a linear and non-linear scaling of a virtual hand vector to extend the reach, enabling interaction at far distances. The method maps hand movements which are inside the natural range of a humans hand in a linear way, whereas out of this range the linear physical displacement of the hand is mapped to an exponentially increasing displacement in the virtual world. For our method, we also aim to extend the reach of the physical hand of the operator. However for our purpose, two separate movement mappings are not suitable as we do not expect interaction with the trajectory in the reach of the operator's hand. Moreover, in preliminary experiments we found that a non-linear mapping decreased the positioning accuracy of the hand. As a result, we utilize linear mapping of the physical hand vector only and scale it from 1 to a predefined maximum value. Throughout placement of the virtual hand in 3D and relative to the trajectory, the operator can indirectly influence (manipulate) its shape. Details about the model of this influence are outlined in the following section.

B. Gravitational Force Field-Based Manipulation

In general, smoothness of a trajectory plays an important role for navigation. This includes use cases where either motion planning is used to generate camera movements along a virtual path, but also in real-world applications where smooth motion of the UAV must be guaranteed. In particular, our method is beneficial for use cases where a sudden change of the situation can occur. Instead of fully halting navigation and globally replanning the trajectory, the operator is able to manipulate it during flight on a local basis. To enable manipulation, we designed a manipulation method based on Newton‘s law of gravitation, whereas the law in general is given with

$$F_G = G \cdot \frac{m_1 \cdot m_2}{r^2} \tag{1}$$

F_G is the resulting gravitational force acting between two objects, m_1 and m_2 are the masses of the two objects influencing each other, r is the distance between the center points of the two masses and G is the gravitational constant. For our indirect manipulation method we adapt this model, resulting in a modified law, given by equation Equ. 2. The two objects are now described by their 3D positions, expressed in world coordinates, and according masses. The position of a coarse trajectory point is expressed by vector $\vec{x}_P(t)$ and position of the hand by vector $\vec{x}_H(t)$ (Sec. III-A). The according masses are given with m_P and m_H. By using the gravitational constant G as a non-constant scaling factor $G(t)$, we let the operator interactively vary the resulting force $\vec{F}_G(t)$. It is applied to the trajectory points, whereas the magnitude of the resulting force can be expressed by the vector norm

$$\left|\vec{F}_G(t)\right| = F_G(t) = G(t) \cdot \frac{m_P \cdot m_H}{\left|\vec{x}_P(t) - \vec{x}_H(t)\right|^2} \tag{2}$$

Applying this force to a trajectory point results in a motion. Direction of the force is expressed with its unit vector $\vec{\hat{F}}_G$.

$$\vec{\hat{F}}_G(t) = \frac{\vec{x}_P(t) - \vec{x}_H(t)}{\left|\vec{x}_P(t) - \vec{x}_H(t)\right|} \tag{3}$$

The displacement $s_P(t)$ over time can then be derived by two times integrating the following simple 2^{nd} order differential equation

$$\ddot{s}_P(t) \cdot m_P = \vec{F}_G(t) \tag{4}$$

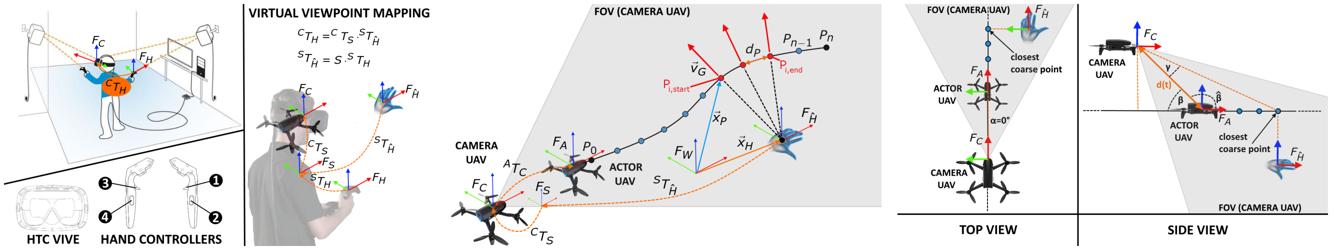
If any dampening force (e.g., friction) is neglected, solving this equation expresses an accelerated movement of each trajectory point in the direction of $\vec{\hat{F}}_G(t)$.

IV. Case Study - UAV Flight Trajectory Manipulation For SAR Missions

Besides of that our novel indirect manipulation method (Sec. III) could be also extended to other applications, with this work we highlight usefulness for UAV flight missions. Our case-study is situated in a virtual SAR scenario where a small town is partly damaged by an earthquake and intervention is necessary. This involves exploration of an outdoor scene with damaged buildings or trees which might collapse or other sudden hazardous situations might occur. We further introduce two UAVs for intervention, whereas we consider the first as main or *ACTOR UAV*. It actually moves along the globally planned trajectories. It could be used for close inspection of objects, like damaged parts of buildings, or carry goods like food or med-kits which are vital for intervention. Moreover, we introduce a second UAV, mainly responsible to improve SA during remote operation. It poses a form of companion UAV, enabling exocentric camera views on the ACTOR UAV, its surrounding scene and its movements along the trajectory. Thus, we call it *CAMERA UAV*.

A. Viewpoint Adaptation

An exocentric viewpoint during navigation along the flight trajectory is vital for effectiveness of our method as it improves the operator's understanding of the scene. Since it is actually provided by an egocentric viewpoint in the camera frame of the CAMERA UAV F_C and supposed to follow the ACTOR UAV, we introduce viewpoint adaptation, affecting ${}^{A}T_C$ (Fig. 2b and Fig. 2c). It is based on the current position of the ACTOR UAV (F_A), current location of the virtual hand ($\vec{x}_H(t)$, $F_{\hat{H}}$) and vertical FOV ($FOV_{HMD,VERT}$)



(a) Hand interaction with HTC Vive.

(b) Basic notations of our method.

(c) Viewpoint adaptation method.

Fig. 2: Overview of our method. a) Depicted is the operators hand interaction from an exocentric viewpoint of the CAMERA UAV (F_C). By utilizing the HTC Vive it enables the indirect force field-based manipulation. b) Basic notations, modelling the hand-based manipulation of the ACTOR UAV's flight trajectory. c) Adaptive viewpoint method based on the current position of the virtual hand in relation to the ACTOR UAV.

of the virtual viewpoint (all notations are expressed in the world frame F_W). While it would have been interesting to investigate on a more extensive adaptive viewpoint method, taking into account additional environmental influences, this work focuses on the indirect manipulation method only. Moreover, the goal was to keep as many other influencing factors constant in the first step.

V. Apparatus and Implementation

In the following, we give an overview of the implemented hardware and software modules for experimental evaluation and the virtual SAR mission, introduced in Sec. IV.

A. Hardware- And Physical-Setup

This work has been developed using standard commodity hardware. Our implementation runs on a desktop system with an i7-CPU@3500Mhz with 8 cores, 32GB of RAM and an NVIDIA GTX980Ti GPU. We use the HTC Vive HMD as immersive VR interface with the standard hand controllers as input devices.

B. Software Framework

For the presented work, the software framework was implemented in ROS [19]. The framework is divided into a backend and a frontend, whereas the backend is an implementation based on the UAV Gazebo simulator *RoTorS* [20] and mainly responsible for the ODE physics simulation [21]. The frontend is based on RViz and responsible for scene visualization and interfacing the HMD. All components of our method are either implemented as ROS nodes or plugins. The behavior of our manipulation method, such as method flavors, force parameters or parameters for the hand interaction, can be also adjusted online by modifying parameters via the callbacks that are exposed by the *dynamic_reconfigure* API of ROS. The source code of our implementation can be freely accessed [22].

C. Scene Visualization

During simulation we visualize several essential components inside actually two different scene models. The first model is a virtual mockup scene where we evaluate the presented manipulation method during a user study. With the second model we present our method for a more specific use case where we simulate disaster intervention during a SAR mission. Essential visualized components include: the two UAVs, trajectories, obstacles, the virtual hand for manipulation, the resulting gravitational force and its influence on the trajectories interpolation points respectively. Please also refer to our supplementary video (see footnote 1).

D. Reconstruction Of The Environment

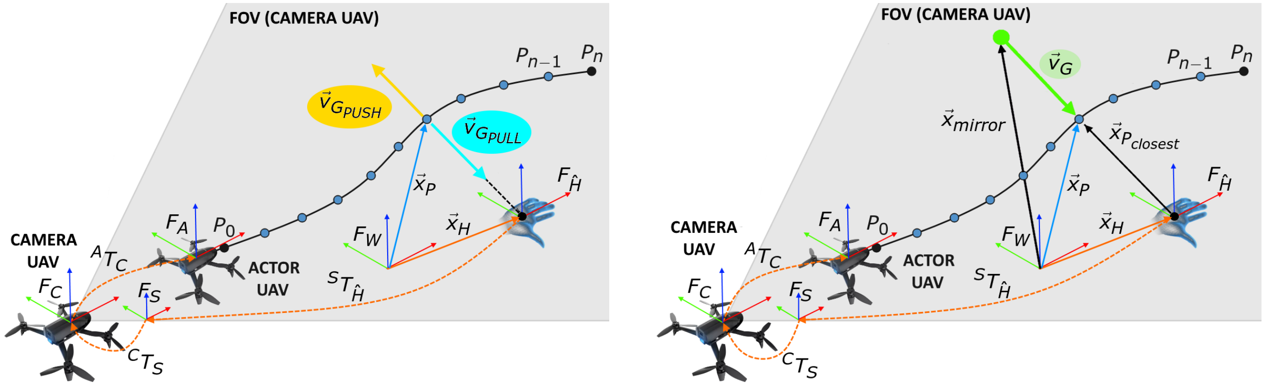
In our simulated scenes, the reconstruction data is represented by 3D occupancy grid maps [23]. We use this occupancy information to generate global feasible paths, which is discussed in the following section. Further, we assume that our scene was fully reconstructed already and geometric information is considered as static for our experiments. We in particular emphasize use-cases where hazardous flight areas have to be avoided but are hard to reconstruct in real-time. Such cases might include gas leaks, striking sparks from a transformer or similar. Although it would have been interesting to investigate on dynamic real-world scenes, as future work a potential lack of geometric information must be addressed beforehand.

E. Path-Planning and Navigation

For generation of flight trajectories we use a global path planning approach based on Probabilistic Road Maps (PRM) [24]. Before navigation we fit a cubic spline through the coarse points of the planned trajectory and further generate a basic velocity profile for a desired constant velocity $\vec{v}_{UAV} = 0.3\frac{m}{s}$. We then feed this data into the tracking controller of Lee [25], included in the *RotorS* framework. For simplicity we control the orientation of the ACTOR UAV so that the x-axis of F_A is always pointing towards the next coarse trajectory point.

F. Viewpoint Adaptation

Our method requires exocentric viewpoints which are adapted as follows (Fig. 2c). We let the CAMERA UAV always *follow* the ACTOR UAV which means for our case-study the CAMERA UAV is always hovering behind and above the ACTOR UAV. We further define horizontal and vertical offset angles α and β, constraining the CAMERA



(a) PUSH-PULL flavor (C1, C2).

(b) MIRROR flavor (C3).

Fig. 3: Manipulation method flavors, representing conditions C1 (PUSH), C2 (PULL) and C3 (MIRROR). Included is the axis- and button-mapping related to our HMD interface, also discussed in Sec. V-A.

UAV in relation to the ACTOR. While we keep $\alpha = 0 = const.$, we define

$$\beta = 45^\circ + (\frac{1}{2} - \frac{1}{3}) \cdot FOV_{HMD,VERT} = const. \tag{5}$$

and introduce time dependent distance d. The distance $d(t)$ is adaptive (Fig. 2c) and depends on the current position of the virtual hand projected onto the virtual trajectory. Finally, d gets adapted so that the ACTOR UAV is always centered in the lower third of a thought fixed viewpoint of the HMDs FOV (constant offset of 45° from the trajectory), resulting in the overall offset angle β, given in Equ. 5. Further, we constrain the angle in terms of how the current position of the hand can be projected into the horizontal FOV of the HMD. We denote this as the angle γ. To conveniently select γ, we relate to the *rule of thirds* [26], which is a framing technique and originated in cinematography [27]. Hence, projecting the virtual hands closest coarse trajectory point into the FOV results in an offset angle γ. For our case-study, we assume that any obstacle avoidance task requires the virtual hand moving in front of the ACTOR UAV. Considering the afore mentioned angle constraints, the distance $d(t)$ can then be defined with

$$d(t) = \frac{d_{hand}(t)}{\sin(\gamma)} \cdot \sin(180^\circ - (\gamma + \hat{\beta})) \tag{6}$$

whereas $d_{hand}(t)$ is the distance from the virtual hands closest coarse trajectory point to the center of the ACTOR UAV, $\gamma = \frac{1}{3} \cdot FOV_{HMD,VERT}$ is the resulting projected angle of the hand in the HMD and $\hat{\beta} = 180^\circ - \beta = const.$ is defined by constant vertical offset angle β (Equ. 5) of the CAMERA UAV when following the ACTOR UAV.

VI. Experimental Evaluation

Considering the indirect manipulation method based on gravitational law, the main focus of this work was to evaluate the most promising flavor of our method in the first step. All designed flavors are described in detail in Sec. VI-B and Fig. 3 and evaluated during a user study. It must be added that a variety of other parameters is influencing our manipulation method, whereas it was not possible to investigate on all of them in our evaluation. This lead us to setting as many parameters and environmental conditions either to optimum values, stated in related work. Or, if no reference was available, the parameter was kept constant in all conditions during our experimental evaluation. Details of the involved parameters and their settings are described in Sec. VI-C.

A. Hand Interaction with HTC-Vive HMD

For hand interaction inside our virtual scenes we use the HTC Vive HMD, also described in Sec. V-A. For our experimental evaluation, we suggest a custom mapping of controller inputs and relative transformations of the hand controllers to the operator's head pose. An overview of the mapping is depicted in Fig. 2a. Based on tracking the operator's physical hand relative to the head (F_H relative to F_C) we can derive the relative transformation ${}^CT_H$. In addition, since directly deriving the hand vector based on ${}^CT_H$ would result in unnatural hand-gestures, we introduce an artificial coordinate frame F_S, located in the shoulder of the operator. During our experiments we define a constant transformation ${}^CT_S$ which reflects the offset of the shoulder frame of the according left or right hand from the operator's head and frame of the CAMERA UAV F_C. Thus, we enable the operator to scale translation of the resulting transformation ${}^ST_{\hat{H}}$ for placing the virtual hand ($F_{\hat{H}}$), which is originated in the shoulder frame F_S. Scaling the physically tracked hand movements to the hand vector in the virtual scene can then be expressed with ${}^ST_{\hat{H}} = S \cdot {}^ST_H$, whereas S is the scaling factor ranging from 1.0 to the maximum hand vector length $|\vec{x}_H|_{max} = S \cdot |\vec{x}_H| = 50.0 \cdot |\vec{x}_H|$ and the rotational component of ${}^ST_H$ is left unchanged. S can be adjusted by the operator with a linear axis of the controller (Fig. 2a, axis ❶).

B. Manipulation Method Flavors

We designed three flavors of our method, whereas they also reflect the 3 study conditions (C1, C2 and C3) for our user study (Sec. VI-D). Based on the method outlined in Sec. III, we adapted the gravitational law as follows.

Applying a resulting gravitational force F_G directly on an influenced trajectory point, results in an accelerated movement. In preliminary experiments we found that this is unintuitive, since during the time the operator applies any force ($t_1 ... t_2$), at time t_2 the point continues moving with the accumulated velocity $v_G(t)$, described with

$$v_G(t) = v_{G0} + \int_{t_1}^{t_2} a_G(t) \cdot dt \quad \text{and} \quad a_G(t) = \frac{F_G(t)}{m_P} \tag{7}$$

v_{G0} is arbitrary and $a_G(t)$ is the resulting acceleration, caused by the gravitational force $F_G(t)$ and applied by the operator at time t. Noteworthy, for all three flavors we keep the involved masses m_H and m_P constant with $m_H = m_P = 1kg$. Further, m_H is not affected by any back influencing forces of the trajectory points. Remarkably, if the operator desires to stop motion, this cannot be controlled directly, as the mass has to be decelerated again, resulting in a continuous time-consuming process, decreasing overall effectiveness of our method. Instead, we let the operator directly control the velocity of the point mass and hence enable to also directly stop motion of the influenced point. The resulting magnitude of this *gravitational velocity* or *influence* $v_G(t)$ is directed along $\vec{F}_G(t)$ (Equ. 3) and applied to the influenced trajectory point, whereas the resulting displacement of the point $s_P(t)$ is then given with

$$s_P(t) = s_0 + \int_{t_1}^{t_2} v_G(t) \cdot dt \quad \text{and} \quad v_G(t) \hat{=} \frac{\left|\vec{F}_G(t)\right|}{m_p} \tag{8}$$

In preliminary experiments we found that the following parameters, describing the gravitational-force-field based interaction, worked best for our experimental evaluation:

- Max. hand vector scale: $S_{max} = 50.0$
- Max. gravitational velocity magnitude: $v_{G,max} = 5.0\frac{m}{s}$

1) PUSH-PULL Flavor (C1, C2): This flavor represents the initial method, that was designed to indirectly influence trajectory points based on gravitational law within 3D hand interaction (Fig. 2b). It is expressed by the velocity vector $\vec{v}_G$ of the influenced points in a so called *push-pull* configuration (Fig. 3a). The push-pull method describes two flavors of the gravitational law whereas only the sign of the resulting influence changes. The flavors are described with

$$PUSH(\mathbf{C1}) : \vec{v}_G(t) \hat{=} \frac{\vec{F}_G(t)}{m_p}, \; PULL(\mathbf{C2}) : \vec{v}_G(t) \hat{=} -\frac{\vec{F}_G(t)}{m_p} \tag{9}$$

whereas $\vec{v}_G(t)$ is the resulting gravitational velocity vector for each flavor and $\vec{F}_G(t)$ is the gravitational force, formerly expressed in Equ. 2 and Equ. 3.

2) Mirror Flavor (C3): As a third flavor (Fig. 3b) we designed the so called MIRROR flavor. Instead of directly creating the influence at the origin of the virtual hand $F_{\hat{H}}$, we mirror the actual influence around the closest unmanipulated trajectory point. The position of the actual origin of the influence can then be described with

$$MIRROR(\mathbf{C3}) : \vec{x}_{mirror}(t) = \vec{x}_H(t) + 2 \cdot (\vec{x}_{P_{closest}} - \vec{x}_H(t)) \tag{10}$$

where $\vec{x}_H(t)$ is the hand vector, $\vec{x}_{P_{closest}}$ is the position vector of the closest unmanipulated coarse trajectory point to the hands position and $\vec{x}_{mirror}(t)$ is the resulting vector pointing to the actual position of where we generate the influence $\vec{v}_G(t)$ in *PUSH* configuration. All vectors are expressed in the world frame F_W.

C. Task Description

The task of our experimental evaluation is focused on obstacle avoidance. During participation in our user study, operators were put in egocentric view of the following CAMERA UAV (F_C) and able to freely move the viewpoint throughout the head movements of the HMD. For the proposed task, they were expected to manipulate a pre-generated trajectory of the ACTOR UAV. The overall goal was to prevent the ACTOR UAV from colliding with the scenes geometry *without the need of stopping during navigation*. The main task was divided into sub-tasks, whereas a detailed description of all relevant and influencing parameters is discussed in the following (please also refer to Fig. 2b and Fig. 2c):

- For experimental evaluation of the indirect manipulation method flavors we use a mockup scene. The scene consists of a simple ground plane, whereas it is textured with a checkerboard pattern. The pattern size of one square is $a = 1.0m$.
- At the beginning of the task, the ACTOR UAV is hovering inside the mockup scene at height $h_{hover} = 1.0m$ and velocity $\vec{v}_{UAV} = 0.0\frac{m}{s}$.
- The UAV is visualized by a real-world scaled model, encapsulated by a sphere with diameter $r_{UAV} = 0.25m$, whereas the blue color provides high contrast against the environment.
- Once the subject is ready and manually confirms start of the experiment (Fig. 2a, button ❷), a straight global path is planned from point $P_A = \{0.0, 0.0, 1.0\}$ to $P_A = \{100.0, 0.0, 1.0\}$, holding 201 coarse points. The radius of visualized points is the radius of the UAV $r_{UAV} = 0.25$, since this is vital for the subject to perceive visual feedback for avoiding a potential collision.
- The UAV then starts moving along the trajectory (trajectory points with interdistance $d_P = 0.5m$) with velocity $\vec{v}_{UAV} = 0.3\frac{m}{s}$. The velocity influences how immediate manipulation has to happen (t_1+t_x). At time t_1 we simulate a sudden change of the environment requiring manipulation of the trajectory and t_x is the duration until the actual collision would happen.
- The sudden change of the environment, based on the obstacles with spherical shape and radius $r_O = 1.0m$. The maximum distance to the center point of obstacle $d_{Omax} = 9.0m$ which results in $t_x = 30s$ given the speed of the ACTOR UAV $\vec{v}_{UAV}$.
- Appearance of obstacles leads to a required manipulation of the trajectory, leading to a deformation D. D can be parametrized with number of points P_i, that are necessary to be influenced and displacement $s_P(t)$ (Equ. 8) of each point leading to a collision free

motion. Participants have to place the hand vector $\vec{v}_H(t)$ and enable the influence (gravitational velocity $\vec{v}_G(t)$) throughout triggering the according axis on the HTC vive controller (Fig. 2a, axis ❸).

- The result is a manipulated trajectory before $t_1 + t_x$ has passed. The operator has to manually confirm a potentially new collision-free trajectory (Fig. 2a, button ❹) in between this time span to complete the task successfully ($t_{conf} \leq t_x$).
- After manipulation, only the locally influenced trajectory points ($P_{i,start}$ to $P_{i,end}$, Fig. 2b) are reoptimized.

After the subject manually confirmed completion of the task, evaluation of the following parameters was conducted, which also reflect our main performance metrics:

- **Task completion time** $t_1 + t_{conf}$, whereas t_1 is constant and t_{conf} is the time after which the operator confirms to update the manipulated trajectory.
- **Trajectory smoothness** after manipulation, whereas we assume a constant amount of interpolation points per flight distance. We calculate a smoothness estimate for the trajectory by accumulating the standard deviations of differences of the displacement of the points ($P_{i,start}$ to $P_{i,end}$) along all 3 dimensions [28] ($\sum \sigma_{diff}$, Fig. 4).

D. User Study

1) Study Design: The main objective of our user study was to assess the effect of the different method flavors (C1, C2 and C3) on the participants task completion times and trajectory smoothness with respect to the obstacle avoidance task. Participants were further asked about their personal preference comparing the three method flavors. Results are reported in Sec. VII. For our study we formulated the following hypothesis:

- H1: Task completion times are lowest in C3.
- H2: Smoothness of the trajectory after manipulation is highest in C3.

To compensate for learning effects, we conducted a training phase before starting the actual evaluation for each condition. We based our study on *within-subject design* and varied the order of the conditions using *full counterbalancing*.

2) Participants: Study participants were either institute members or students, working in the field of computer science. We invited 6 males between the ages of 23 and 30 ($m = 26.25$, $\sigma = 3.304$). Participation in the study was completely voluntary. 2 of the participants did not finish the user study due to environmental influences. As a consequence, their results were excluded.

3) Analysis: In each of our 4 sessions, we tested indirect manipulation of a flight trajectory in all three conditions, reflecting evaluation of the individual method flavors (Sec. VI-B). With each flavor participants repeated the task with 4 obstacles, resulting in a total of 48 valid runs for the obstacle avoidance task. For each participant, we measured task completion times and smoothness of the manipulated trajectories. In addition, participants were asked to fill out a NASA-TLX [29] questionnaire (Scale: 0-100), as well as a custom questionnaire with respect to their experience in the respective condition. The custom questionnaire contained 7-point Likert items, ranging from 1 (”strongly disagree”) to 7 (”strongly agree”). We asked participants about task completion confidence and comfort, as well as their perception of accuracy during manipulation. We report interval estimates for a 95% confidence interval (CI).

VII. Results

Based on mean and CI (see Fig. 4) in this section we discuss results of our user study. All participants mentioned that they felt comfortable with the method flavors C1 and C3, in contrast to C2. This is also supported by the perceived accuracy during manipulation. Further, all participants were able to avoid the given obstacles without collisions which is also reflected in the task completion confidence.

Remarkably, C1 achieved lowest task completion times and highest trajectory smoothness, rejecting H1 and H2. However, 2 participants commented on that they felt limited in terms of where to conveniently place the force influence and predicting where the trajectory would move when pushing. With these regards, participants confirmed that C3 felt more natural and intuitive since the trajectory directly converges towards the influencing force. It must be added, that C3 would enable the operator to manipulate the path with placing the *visualized influencing force* outside of any potential hazardous area. The actual influence could be then still generated inside, while enabling smooth manipulation. With these regards, the *PULL* flavor is clearly not applicable for the suggested use case, as it would commonly result in non-optimal shapes. If this flavor is applied for longer time periods $(t_2 - t_1) >>$, all influenced trajectory points converge towards the position of the source of the influencing force. This is also supported by the participants comments as all report oversensitive and unnatural behaviour and shaggy or messy trajectory shapes after manipulation.

VIII. Conclusion and Future Work

For a variety of use cases, like SAR missions, navigation along UAV flight trajectories plays an important role. Without the SA of the remote operator, the UAV may reach hazardous areas of the scene. Ultimately, this could lead to that the overall task performance is negatively affected or the mission fails if the UAV gets damaged. In this paper, we propose a novel indirect manipulation method to resolve such cases more effectively. We allow the operator to quickly manipulate the trajectory, on a local basis, at far distances. Thus, we avoid global replanning and fully stopping the UAV. In a first step we were able to investigate on the most effective flavors of our method, based on gravitational law, whereas we kept as many other environmental influences and parameters constant during our experiments. While we have highlighted effectiveness of two flavors (C1 and C3) in simulation, as future work we would like to investigate on our method as part of a more extensive navigation interface for dynamic real-world scenes.

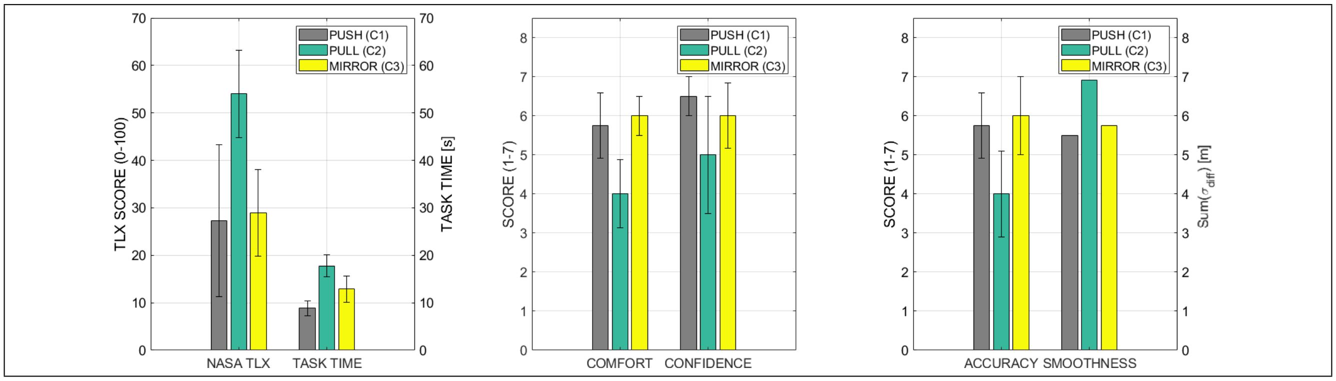


Fig. 4: Results of our user study, indicating decreasing NASA-TLX score with the PUSH (C1) and MIRROR (C3) flavors. Interestingly, C1 achieved lowest task completion times and highest trajectory smoothness, whereas, in contrast, 2 participants commented on higher mental demands regarding limited placement of the influence. C2 was found to be most demanding.

References

[1] Jessie YC Chen, Ellen C Haas, and Michael J Barnes. Human performance issues and user interface design for teleoperated robots. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 37(6):1231–1245, 2007.

[2] Scott Frees and G Drew Kessler. Precise and rapid interaction through scaled manipulation in immersive virtual environments. In *Virtual Reality, 2005. Proceedings. VR 2005. IEEE*, pages 99–106. IEEE, 2005.

[3] Seyedkoosha Mirhosseini, Ievgeniia Gutenko, Sushant Ojal, Joseph Marino, and Arie E Kaufman. Automatic speed and direction control along constrained navigation paths. In *2017 IEEE Virtual Reality (VR)*, pages 29–36. IEEE, 2017.

[4] Doug A. Bowman and Larry F. Hodges. An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. In *Proceedings of the 1997 Symposium on Interactive 3D Graphics*, I3D '97, pages 35–ff., New York, NY, USA, 1997. ACM.

[5] Keita Higuchi and Jun Rekimoto. Flying head: a head motion synchronization mechanism for unmanned aerial vehicle control. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems*, pages 2029–2038. ACM, 2013.

[6] David Mirk and Helmut Hlavacs. Using drones for virtual tourism. In *International Conference on Intelligent Technologies for Interactive Entertainment*, pages 144–147. Springer, 2014.

[7] John Paulin Hansen, Alexandre Alapetite, I Scott MacKenzie, and Emilie Møllenbach. The use of gaze to control drones. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, pages 27–34. ACM, 2014.

[8] John Thomason, Photchara Ratsamee, Kiyoshi Kiyokawa, Pakpoom Kriangkomol, Jason Orlosky, Tomohiro Mashita, Yuki Uranishi, and Haruo Takemura. Adaptive view management for drone teleoperation in complex 3d structures. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces*, pages 419–426. ACM, 2017.

[9] Ravin Balakrishnan and Gordon Kurtenbach. Exploring bimanual camera control and object manipulation in 3d graphics interfaces. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 56–62. ACM, 1999.

[10] Nicolas Burtnyk, Azam Khan, George Fitzmaurice, and Gordon Kurtenbach. Showmotion: camera motion based 3d design review. In *Proceedings of the 2006 symposium on Interactive 3D graphics and games*, pages 167–174. ACM, 2006.

[11] D Mendes, FM Caputo, A Giachetti, A Ferreira, and J Jorge. A survey on 3d virtual object manipulation: From the desktop to immersive virtual environments. In *Computer Graphics Forum*. Wiley Online Library, 2018.

[12] Daniel Mendes, Daniel Medeiros, Maurício Sousa, Eduardo Cordeiro, Alfredo Ferreira, and Joaquim A Jorge. Design and evaluation of a novel out-of-reach selection technique for vr using iterative refinement. *Computers & Graphics*, 67:95–102, 2017.

[13] B. Fowler and R. Bartels. Constraint-based curve manipulation. *IEEE Computer Graphics and Applications*, 13(5):43–49, Sept 1993.

[14] Paul Michalik, Dae Hyun Kim, and Beat D Bruderlin. Sketch-and constraint-based design of b-spline surfaces. In *Proceedings of the seventh ACM symposium on Solid modeling and applications*, pages 297–304. ACM, 2002.

[15] Carlo Masone, Paolo Robuffo Giordano, Heinrich H Bülthoff, and Antonio Franchi. Semi-autonomous trajectory generation for mobile robots with integral haptic shared control. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6468–6475. IEEE, 2014.

[16] Carlo Masone, Mostafa Mohammadi, Paolo Robuffo Giordano, and Antonio Franchi. Shared planning and control for mobile robots with integral haptic feedback. *The International Journal of Robotics Research*, 37(11):1395–1420, 2018.

[17] Zhan Gao and Anatole Lécuyer. Path-planning and manipulation of nanotubes using visual and haptic guidance. In *VECIMS*, 2009.

[18] Ivan Poupyrev, Mark Billinghurst, Suzanne Weghorst, and Tadao Ichikawa. The go-go interaction technique: non-linear mapping for direct manipulation in vr. In *Proceedings of the 9th annual ACM symposium on User interface software and technology*, pages 79–80. ACM, 1996.

[19] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, Japan, 2009.

[20] Fadri Furrer, Michael Burri, Markus Achtelik, and Roland Siegwart. Rotorsa modular gazebo mav simulator framework. In *Robot Operating System (ROS)*, pages 595–625. Springer, 2016.

[21] Russell Smith et al. Open dynamics engine. 2005.

[22] W. A. Isop. Indirect flight trajectory manipulation (Github repository). `https://github.com/weraleiso/indirect_flight_trajectory_manipulation`. Accessed: 2019-02-01.

[23] Kai M Wurm, Armin Hornung, Maren Bennewitz, Cyrill Stachniss, and Wolfram Burgard. Octomap: A probabilistic, flexible, and compact 3d map representation for robotic systems. In *Proc. of the ICRA 2010 workshop on best practice in 3D perception and modeling for mobile manipulation*, volume 2, 2010.

[24] Lydia E Kavraki and Jean-Claude Latombe. Probabilistic roadmaps for robot path planning. *Citeseer*, 1998.

[25] Taeyoung Lee, Melvin Leok, and N Harris McClamroch. Control of complex maneuvers for a quadrotor uav using geometric methods on se (3). *arXiv preprint arXiv:1003.2005*, 2010.

[26] Media College. Framing. `https://www.mediacollege.com/video/camera/tutorial/01-framing.html`. Accessed: 2018-11-14.

[27] Daniel Arijon. Grammar of the film language. *Silman-James Press*, 1976.

[28] Nelson David Munoz Ceballos, Jaime Alejandro Valencia, and Nelson Londono Ospina. Quantitative performance metrics for mobile robots navigation. In *Mobile Robots Navigation*. InTech, 2010.

[29] Sandra G Hart and Lowell E Staveland. Development of nasa-tlx (task load index): Results of empirical and theoretical research. In *Advances in psychology*, volume 52, pages 139–183. Elsevier, 1988.