

# Shared Control for Game Accessibility: Understanding Current Human Cooperation Practices to Inform the Design of Partial Automation Solutions

DRAGAN AHMETOVIC, Università degli Studi di Milano, Dipartimento di Informatica, Italy

MATTEO MANZONI, Università degli Studi di Milano, Dipartimento di Informatica, Italy

FILIPPO CORTI, Università degli Studi di Milano, Dipartimento di Informatica, Italy

SERGIO MASCETTI, Università degli Studi di Milano, Dipartimento di Informatica, Italy

Shared control is a form of video gaming accessibility support that allows players with disabilities to delegate inaccessible controls to another person. Through interviews involving 14 individuals with lived experience of accessible gaming in shared control, we explore the ways in which shared control technologies are adopted in practice, the accessibility challenges they address, and how the support currently provided in shared control can be automated to remove the need for a human assistant. Findings indicate that shared control is essential for enabling access to otherwise inaccessible games, but its reliance on human support is a key limitation. Participants welcomed the idea of automating the support with software agents, while also identifying limitations and design requirements. Accordingly, this work contributes insights into current practices and proposes guidelines for developing automated support systems.

CCS Concepts: • **Applied computing** → *Computer games*; • **Human-centered computing** → *Accessibility technologies*; **Collaborative interaction**; **Empirical studies in accessibility**; • **Social and professional topics** → *People with disabilities*.

Additional Key Words and Phrases: Shared control; partial automation; human cooperation; accessible gaming.

## 1 Introduction

Video games have become a leading entertainment industry, with more than 3.35 billion video game players worldwide [52] (41% of the global population). Among them, 20% have some form of disability. Furthermore, 46% of people with disabilities report playing video games regularly [45].

Numerous assistive technologies have been developed to enable autonomous access to video games by players with disabilities [3]. Despite these technologies, not all players are able to access every game independently [7, 39]. To address this issue, *Shared Control* solutions have been developed, such as *Xbox Controller Assist* [41], allowing players with disabilities to delegate control of certain game actions to another person. Recent research also explores how to extend this approach by replacing human support with a software agent [12]. However, to the best of our knowledge, no prior study investigates how existing shared control technologies are commonly used by players, to address which accessibility challenges, and how these needs can be translated into technologies that automate such support.

To address these issues, we conduct a study, through interviews and focus groups, with 14 individuals who have prior experience in the use of shared control systems, including people with disabilities who play using shared control, others who support players with disabilities during shared control, and experts in assistive gaming technologies. Through reflexive thematic analysis, the research first aims to gain a deeper understanding of the shared control technologies currently in use, highlighting their benefits and limitations. Second, we investigate the acceptability and the key features that a system automating the role of the supporting person should have.

---

Authors' Contact Information: Dragan Ahmetovic, dragan.ahmetovic@unimi.it, Università degli Studi di Milano, Dipartimento di Informatica, Milan, Italy; Matteo Manzoni, matteo.manzoni@unimi.it, Università degli Studi di Milano, Dipartimento di Informatica, Milan, Italy; Filippo Corti, filippo.corti1@studenti.unimi.it, Università degli Studi di Milano, Dipartimento di Informatica, Milan, Italy; Sergio Mascetti, sergio.mascetti@unimi.it, Università degli Studi di Milano, Dipartimento di Informatica, Milan, Italy.

Results show that shared control is essential for making certain games accessible that would otherwise not be playable. However, these techniques have a major limitation: they require the availability of a person willing to provide support. For this reason, participants welcomed the possibility of replacing the supporting person with a software agent, while at the same time highlighting both the potential limitations of the solution and the features it should possess. Our work, therefore, provides a better understanding of the current use of shared control systems and outlines design guidelines for the development of shared control technologies with automated support.

## 2 Background

This section surveys the accessibility problems faced by gamers with disabilities in playing video games (Section 2.1), and the existing assistive technologies designed to overcome these limitations (Section 2.2). Then an overview of the state of the art in shared control solutions is presented, both in general contexts (Section 2.3) and specifically when applied to gaming (Sections 2.3.1 and 2.3.2).

### 2.1 Video Games Accessibility

Playing video games poses accessibility challenges for people with disabilities [3, 6, 8, 25, 39, 64]. These challenges include single-sense feedback (*e.g.*, audio cues without visual counterparts) [6, 8, 25, 39], unsuitable button layouts [8, 18, 39], and demanding input gestures, such as repeated button mashing or long holds [6, 8]. Thus, after identifying a game that they find interesting, gamers with disabilities have to adapt the game to their specific needs [39]. In some cases, however, adaptations fall short, forcing players to “play their own game”, engaging it in unconventional ways [25], or even to abandon the game altogether [39]. Guidelines written to facilitate the development of in-game accessibility features exist [6, 7, 25, 39]. However, these guidelines are rarely implemented [3, 64]. Due to this, accessibility often needs to be achieved through outside-of-game solutions. For example, participants in our interview often mentioned accessible game controllers and input-modifying software.

### 2.2 Accessibility of Video Game Controls

Various peripherals are used as video game controllers. In particular, gamepads provided with major gaming consoles have evolved to share several common design features [36, 37]. For example, Xbox and PlayStation controllers are designed to be held with both hands and require coordinated use of the thumbs (each capable of controlling a joystick and four buttons) as well as two additional fingers (usually the index and middle fingers) to operate two side buttons per hand. This design assumes that all users have similar hand anatomy and size [9], can operate multiple inputs with both hands [18], and are able to perform quick, coordinated, and reactive movements [18]. As a result, standard controllers may be inaccessible to players with disabilities, for example, those with upper-limb mobility impairments.

Alternative controllers, designed to be more accessible and customizable, offer different button layouts from standard ones. Some are designed to be operated with a single hand [62] or placed on a flat surface [1, 23, 43]. Others are personalizable through external buttons [17, 23, 43] or a modular design [10].

Software support has also been proposed to enhance the accessibility of game controllers. Some tools allow remapping controller buttons to different commands; these are sometimes available directly within the games themselves. Others allow remapping of commands across different types of input devices. For example, JoyToKey [31] allows a game controller to emulate keyboard inputs for games designed to be played with a keyboard. Further solutions enable users to perform actions using voice input [4, 60] or facial expressions [38, 51], which can then be mapped to specific game commands.

Through the use of these tools, inaccessible games can be made accessible to many people with disabilities. The resulting game setup, that is the specific configuration of hardware and software accommodations and accessibility tools used, can be quite articulated, complex, and can vary based on the game, the disability, and the preferences of the player. However, for some gamers with disabilities, existing accessibility tools may not suffice to access and effectively use all the inputs required to play a game [12]. In such cases, one possibility is to delegate inaccessible game controls to someone else through shared control [14, 41].

### 2.3 Shared Control

Shared control refers to two or more agents collaboratively interacting with a system. Beyond video games, shared control has been studied for collaborative robot control [21, 27], assisted vehicle driving [34], and management of shared user interfaces [20]. In these contexts, the proposed interaction models involve either cooperation between humans or the integration of software agents to assist the user.

The terminology used to distinguish between different forms of support varies across application domains. For example, when referring to cooperation between only human actors, commonly used terms include *human-human collaboration* [61], *multi-user interaction* [20], and *multi-operator-single-robot* [24]. Similarly, systems in which software agents contribute to the control are referred to using terms such as *partial automation* [14], *human-AI collaboration* [61], *human-in-the-loop* [27], or *supervisory control* [5].

To disambiguate the various meanings, we use the term *shared control* to refer to all interactions that allow game controls to be distributed across multiple actors, usually a *pilot*, i.e., the person primarily responsible for playing (typically a person with disabilities), and a *copilot* supporting the pilot. As illustrated in Figure 1, the two subtypes of shared control are: *human cooperation*, when the copilot is a human actor; and *partial automation*, when the copilot is a software agent.

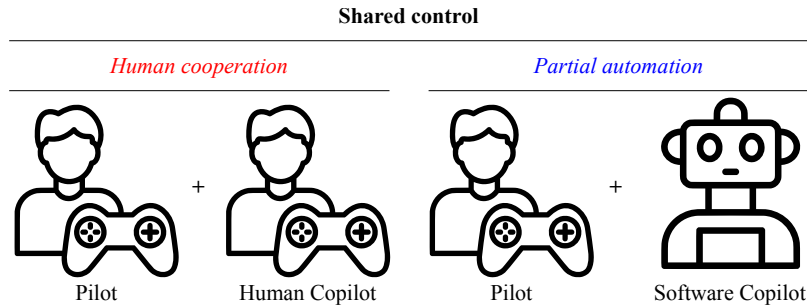


Fig. 1. Relationship between shared control, human cooperation, and partial automation

**2.3.1 Human Cooperation for Video Games Accessibility.** Several human cooperation solutions have been proposed in the context of video games, although not explicitly as assistive technologies. Instead, most studies have focused on analyzing how different forms of cooperation affect aspects such as social interaction, gameplay experience, and perceived enjoyment. Works by Loparev [35], Sykownik [56], and Rozendaal [53] observed that introducing interdependence mechanics between players led to higher levels of social interaction compared to independent play. Conversely, Rozendaal et al. [53] also noted that requiring cooperation to progress through the game could reduce players' perceived autonomy and control.

In the context of video game accessibility, Microsoft has released *Xbox Controller Assist* (formerly *Xbox Copilot*) [41], a software available on Xbox consoles and Windows PCs that allows linking two controllers so that two players can use them to control the same game actions, as if they were a single controller. This solution is actively advertised as an accessibility feature “to help a friend or loved one through a game or console experience”, and evidence of the usage of this solution by players with disabilities is available on various social media<sup>1</sup>. Similarly, on *PlayStation 5* (PS5) consoles, the *PS5 Access Controller* [23] can be paired with another PS5 controller, enabling human cooperation. A third-party solution, the *Titan Two* adapter [16], supports remote human cooperation over the internet. In this configuration, the game feedback (video and audio) is shared with a remote player who can play on their controller. Their input is then transmitted to the local console or PC, where it is merged with the local player’s input, enabling collaborative control of the same game session even when players are not physically in the same location.

Gonçalves et al. have investigated human cooperation in relation to video game accessibility [26]. Specifically, they developed two video games intended to be played in pairs by individuals with mixed abilities: one sighted player would face a visual challenge, while one blind player would face an auditory one. Their findings demonstrate that cooperation between players with differing abilities is not only possible but can be effectively supported through game design that leverages each participant’s strengths. However, the authors themselves acknowledge that this form of cooperation does not constitute a generalizable accessibility solution, as it relies on custom-designed games and does not apply to disabilities beyond visual impairments.

**2.3.2 Partial Automation for Video Games Accessibility.** Partial automation is a type of shared control that removes the necessity for human copilots, substituting them with a software agent. In the video game accessibility domain, partial automation has been implemented in commercial video games [2, 22, 49, 50, 55, 58], and it has been explored in prior research as well [11–13, 28, 30, 59]. However, the application of partial automation has been mostly limited to specific game tasks. One use case is player balancing: providing assistance in a given task that is inversely proportional to the player’s skill level, thus ensuring a similar gaming experience for all. For example, some first-person shooters such as *Call of Duty* [2] integrate aim-assistance, whose effectiveness has been evaluated in prior research [30, 59]. Similarly, some racing games implement steering assistance to help players stay on the road [11, 22]. Another approach is to reduce the effort required to play by automating some inputs. For example, some racing games such as *Forza Motorsport* [58] implement automatic gear shifting. More pronounced implementations of partial automation are found in games like the *Bayonetta* series [49, 50], which feature an automatic mode where the player only needs to focus on attacking while the game autonomously handles movement and enemy targeting. Another example is *Zac - O Esquilo* [40], a one-switch (controlled with only one button) video game in which the player decides when to move the avatar, but an algorithm determines the direction of the movement. These solutions are specifically designed and implemented for each game, but lack generalizability, which is a significant barrier to the widespread adoption of partial automation as a solution for video game accessibility.

Cimolino et al. [12] employ partial automation in two research video games to demonstrate how distributing gameplay actions between a player and a software agent can enable players with motor disabilities to play. In this approach, the player performs only the actions that they are physically capable of, while the remaining actions are delegated to the software agent. Findings from the study highlight the importance of mutual understanding between the player and the agent. Lack of mutual understanding may lead to *automation confusion* [13], a phenomenon in which the player struggles to distinguish the outcome of their own actions from those performed by the agent.

<sup>1</sup>A list of social network posts on this topic is available as supplemental material.

These works contribute to a better understanding of how partial automation can support video game accessibility. However, to the best of our knowledge, no prior work explores existing shared control practices, how they are used, and with what purposes. Most importantly, it is unclear how the support currently provided through human cooperation can be provided through generalizable partial automation solutions and how such solutions should be designed.

### 3 Methodology

To understand how human cooperation technologies are currently used by people with disabilities, and to inform the design of future partial automation approaches, semi-structured interviews and focus groups [33] were organized with people who regularly use, or have used in the past, human cooperation systems for video games. The experimentation was approved by the Ethics Committee of the University of Milan (opinion no. 16/25).

#### 3.1 Participants

Participants were recruited through announcements on social media pages frequented by people with disabilities, such as *r/disabledgamers* on Reddit, and by reaching out to accessibility experts' organizations. A local association for people with motor impairments, Spazio Vita<sup>2</sup> of Niguarda Hospital in Milan - Italy, also supported the recruitment by mediating contact with some participants. The following inclusion criteria were adopted:

- Being of legal age.
- Being able to speak Italian or English.
- One of the following:
  - **(Pilot)** Using or having used human cooperation in the past.
  - **(Copilot)** Using or having used human cooperation in the past to support a person with disabilities.
  - **(Expert)** Being an accessibility expert who uses human cooperation as a tool to help people with disabilities in playing video games.

We recruited 14 participants, among whom 8 were pilots, 5 copilots, and 6 experts. Participants' full demographic profiles are reported in Table 1. For organizational reasons, it was not possible to collect demographic data for **P8<sub>PC</sub>**. One additional participant was excluded from the results because they used Xbox Controller Assist [41] not in human cooperation, but to play alone with two controllers. In total, 4 individual interviews and 4 focus groups were conducted:

- Individual interviews:
  - **P1<sub>P</sub>**: tried but no longer uses human cooperation, having found a gaming setup for playing independently.
  - **P2<sub>P</sub>**: uses remote human cooperation to play with people met online.
  - **P5<sub>CE</sub>**: accessibility professional with experience in hardware solutions for accessible gaming.
  - **P6<sub>E</sub>**: accessibility expert with occasional gaming experience.
- Focus groups:
  - **P3<sub>P</sub>** and **P4<sub>C</sub>**: **P4<sub>C</sub>** is a family member, caregiver, and habitual copilot of **P3<sub>P</sub>**.
  - **P7<sub>PC</sub>**, **P8<sub>PC</sub>**, **P9<sub>PC</sub>**, and **P10<sub>PC</sub>**: members of a weekly gaming group in which they play cooperatively. **P7<sub>PC</sub>** usually pairs with **P8<sub>PC</sub>**, while **P9<sub>PC</sub>** plays with **P10<sub>PC</sub>**.
  - **P11<sub>CE</sub>** and **P12<sub>CE</sub>**: coordinators of the gaming group involving **P7<sub>PC</sub>**, **P8<sub>PC</sub>**, **P9<sub>PC</sub>** and **P10<sub>PC</sub>**. They support the participants both as copilots and by helping to find suitable game setups.

<sup>2</sup><https://spaziovitaniguarda.it>

- **P13<sub>E</sub>** and **P14<sub>E</sub>**: colleagues who assist people with disabilities in identifying and configuring accessible game setups.

Table 1. Participants' Demographic Data. The participant's role is in ID's subscript: P - Pilot, C - Copilot, E - Expert

ID	Age	Gender	Gameplay frequency	Disability		Difficulties in gameplay	Preferred platform
				Type	Onset		
<b>P1<sub>P</sub></b>	29-38	M	Daily	Paresis	Birth	Moderately	Switch, PC, PS
<b>P2<sub>P</sub></b>	29-38	M	Daily	Blindness	Birth	A lot	PC, PS
<b>P3<sub>P</sub></b>	29-38	F	Daily	Spastic quadriplegia	Birth	A lot	Xbox
<b>P4<sub>C</sub></b>	39-50	M	Daily	None	None	None	Smartphone, Xbox
<b>P5<sub>CE</sub></b>	39-50	M	Daily	Burn injury	20 years	A little	Xbox
<b>P6<sub>E</sub></b>	29-38	M	Monthly	None	None	None	Xbox
<b>P7<sub>PC</sub></b>	18-28	M	Daily	Spastic tetraparesis	Birth	Moderately	Switch, PS5
<b>P8<sub>PC</sub></b>	Data Not Available			Yes		Data Not Available	
<b>P9<sub>PC</sub></b>	39-50	F	Daily	Pediatric tetraparesis	Birth	Moderately	Smartphone
<b>P10<sub>PC</sub></b>	18-28	F	Daily	Reduced arm mobility	Birth	A little	Tablet
<b>P11<sub>CE</sub></b>	39-50	M	Weekly	None	None	None	PC, Xbox
<b>P12<sub>CE</sub></b>	39-50	M	Weekly	None	None	None	PC
<b>P13<sub>E</sub></b>	29-38	F	Monthly	None	None	None	Switch
<b>P14<sub>E</sub></b>	39-50	M	Weekly	None	None	None	Switch, Xbox

Ten participants identified as male, while four identified as female. The most represented age groups were 29–38 and 39–50 (6 participants each), followed by 18–28 age group (2 participants). Eight participants declared having a disability, among which **P2<sub>P</sub>** is the only one with a visual impairment, while the others have motor disabilities. The remaining six participants do not have disabilities, but have experience in supporting people with disabilities in using human cooperation. In particular, **P5<sub>CE</sub>**, **P6<sub>E</sub>**, **P11<sub>CE</sub>**, **P12<sub>CE</sub>**, **P13<sub>E</sub>**, and **P14<sub>E</sub>** professionally help people with disabilities in finding hardware and software configurations for gaming according to their needs.

All participants play video games at least once a month. Among those with disabilities, only **P5<sub>CE</sub>** and **P10<sub>PC</sub>** reported having little difficulty in playing video games; the others declared having at least moderate difficulty. The most used gaming platform is Xbox (6 participants), followed by PC and Nintendo Switch (4 participants each), PlayStation (3 participants), smartphone (2 participants each), and tablet (1 participant). Nine people attribute their reference to the ease of access of the platform (**P1<sub>P</sub>**, **P2<sub>P</sub>**, **P3<sub>P</sub>**, **P4<sub>C</sub>**, **P6<sub>E</sub>**, **P9<sub>PC</sub>**, **P10<sub>PC</sub>**, **P11<sub>CE</sub>**, **P14<sub>E</sub>**). Other motivations include familiarity with it (**P4<sub>C</sub>**, **P5<sub>CE</sub>**, **P12<sub>CE</sub>**) and the presence of exclusive video games for the platform (**P7<sub>PC</sub>**, **P13<sub>E</sub>**). Eight of the nine participants with disabilities also use hardware or software tools to facilitate access to video games. These include the Xbox Adaptive Controller (**P3<sub>P</sub>**, **P7<sub>PC</sub>**, **P8<sub>PC</sub>**, **P9<sub>PC</sub>**, **P10<sub>PC</sub>**), custom buttons and joysticks (**P7<sub>PC</sub>**, **P9<sub>PC</sub>**, **P10<sub>PC</sub>**) and software for remapping controller and keyboard keys (**P1<sub>P</sub>**, **P8<sub>PC</sub>**).

### 3.2 Procedure

The interviews were conducted online, using video-calling platforms, and were audio-recorded to facilitate the following analysis. In addition to the participant, at least two members of the research group were always present during each interview: one with the role of main interviewer and the other with a support function, tasked with intervening with additional questions not initially planned. Each interview lasted between 60 and 90 minutes. In cases where users knew

each other and shared usage modalities of human cooperation systems, the interviews were organized as focus groups, in which multiple users participated in the meeting and also discussed among themselves on the proposed topics.

The initial interview outline<sup>3</sup> drew inspiration from prior works on shared control and partial automation [13–15, 26, 53]. The first part of each interview was dedicated to human cooperation technologies. With pilots and copilots, we explored how they used these technologies, which ones they preferred, and what advantages and limitations they had. With accessibility experts, we investigated how they support users who come seeking a new gaming setup tailored to their needs. We then proceeded to understand how they integrate human cooperation technologies into these setups.

In the second part, a concept of a partial automation system for video game accessibility was presented, and interviewees were asked for their opinion, how they would use the system, and what requirements it should have. The system was described as a shared control solution, similar to Xbox Controller Assist [41], but designed to allow playing without the need for a second person. We indicated that the users would be able to select which actions they wanted to control, while the system would autonomously control the remaining ones. The description of the system was purposefully kept vague to encourage the elicitation of its potential functionalities. The initial questions were expanded based on the outcome of the first two interviews. In particular, two additional topics on human cooperation were introduced: communication with the copilot during gameplay and reasons for not using this technology.

### 3.3 Data Analysis

The audio-recordings of the interviews were transcribed and analyzed following the reflexive thematic analysis methodology [57]. The first interview was analyzed jointly by four researchers, identifying initial codes. Subsequent interviews were randomly divided among the researchers. Each interview was coded by one researcher and subsequently reviewed by a second researcher, who integrated any additional observations. In a subsequent meeting, all codes were reviewed and discussed as a group to resolve any ambiguities or disagreements, re-examining the relevant parts of the transcriptions until consensus was reached. Finally, the extracted codes were consolidated and organized into sub-themes and main themes through an iterative process of comparison among researchers.

## 4 Results

Six main themes were identified (Table 2): *Shared Control Benefits and Limitations* (Section 4.1), *Human Cooperation Limitations Addressed in Partial Automation* (Section 4.2), *Copilot’s Interventions* (Section 4.3), *Negotiating Collaboration* (Section 4.4), *Interaction* (Section 4.5), and *Factors Affecting the Collaboration* (Section 4.6). We observe that, although the interviews focused on two distinct areas — human cooperation technologies and the proposal of a partial automation system — we unify their thematic analysis, treating the two areas orthogonally with respect to the identified themes when possible. This choice is motivated by the similarity of the collected codes, which are largely transversal to both areas.

### 4.1 Shared Control Benefits and Limitations

This section explores the benefits of the shared control technology as an accessibility tool (Section 4.1.1), its impact on sociality and inclusion (Section 4.1.2), and ethical concerns related to the misuse of this technology (Section 4.1.3).

#### 4.1.1 Accessibility.

<sup>3</sup>The initial interview outline is available as supplemental material.

Table 2. Themes and sub-themes identified through the reflexive thematic analysis

Themes	Sub-themes
1. Shared Control Benefits and Limitations	1. Accessibility 2. Sociality and Feeling of Inclusion 3. Ethical Concerns
2. Human Cooperation Limitations Addressed in Partial Automation	1. Loss of Autonomy 2. Copilot Availability 3. Copilot's Engagement During Play
3. Copilot's Interventions	1. Assistance During Game Setup 2. Assistance with Menu Access 3. Assistance by Playing 4. Assistance by Signaling 5. Assistance by Suggesting
4. Negotiating Collaboration	1. Actions Separation 2. Policies Guiding Action Assignment 3. Leadership Management 4. Copilot's Operational Autonomy
5. Interaction	1. Verbal and Non-Verbal Communication 2. Intent Understanding
6. Factors Affecting the Collaboration	1. Knowledge of the Game 2. Relationship Between Pilot and Copilot

**Human cooperation.** This approach enables people with disabilities to access potentially any game (**P2<sub>P</sub>**) that is not accessible using other assistive technologies (**P2<sub>P</sub>**, **P3<sub>P</sub>**, **P7<sub>PC</sub>**, **P8<sub>PC</sub>**, **P9<sub>PC</sub>**, **P10<sub>PC</sub>**). It is also beneficial for those games that can be played autonomously, but at a cost of a high physical or cognitive load (**P7<sub>PC</sub>**). **P5<sub>CE</sub>**, **P6<sub>E</sub>**, **P13<sub>E</sub>**, and **P14<sub>E</sub>**, who provide support to people with disabilities in setting up games and devising accessible game configurations, confirm that human cooperation is indeed useful as an assistive solution, albeit with some limitations (Section 4.2). As observed by **P5<sub>CE</sub>**, playing with human cooperation can be initially frustrating, but it may be the only solution to play certain games.

**P5<sub>CE</sub>**: *The psychosocial effect is that there is going to be frustration, but there's also going to be a lot of joy. So you either can't play at all, or you play with a copilot, and you guys learn and grow together.*

**Partial automation.** The idea of introducing partial automation as an alternative to human cooperation was acclaimed by most participants, who recognize its potential as an assistive technology. **P1<sub>P</sub>** considers that, similarly to human cooperation, partial automation would be useful to access games too complex to be played autonomously. **P2<sub>P</sub>**, **P7<sub>PC</sub>**, **P8<sub>PC</sub>**, **P9<sub>PC</sub>**, and **P10<sub>PC</sub>** furthermore highlight that partial automation could be particularly useful in multiplayer games, which commonly have fewer accessibility options. **P7<sub>PC</sub>** and **P10<sub>PC</sub>** would be eager to test this solution. For **P5<sub>CE</sub>**, partial automation could also be effective for users with very low mobility: in such cases, the user could delegate most interactions to the system, while still preserving the decisional role on the progress of the game. That would make the gaming experience more similar to an interactive movie, preserving the narrative content and the active role of the player.

#### 4.1.2 Sociality and Feeling of Inclusion.



**Human cooperation.** Many participants pointed out that human cooperation is not simply about being able to play, but it's also about socializing through playing together. For this reason, **P5<sub>CE</sub>** strongly advocates for human cooperation, even when other accessibility options are available.

**P5<sub>CE</sub>:** *And so, yeah, I think that there is a lot of good psychosocial component to playing together. And when you play alone, it's ok... It's just I think it's better with other people.*

**P14<sub>E</sub>** noted that playing through human cooperation can also strengthen the bond with the copilot, for instance, when the latter is a family member. That's why, when working with children, **P14<sub>E</sub>** usually recommends human cooperation as a solution. Furthermore, **P7<sub>PC</sub>**, **P8<sub>PC</sub>**, **P9<sub>PC</sub>**, and **P10<sub>PC</sub>**, who participate together in weekly gaming workshops (Section 3.1), reported that these meetings are also an opportunity to socialize. **P3<sub>P</sub>** and **P14<sub>E</sub>** further explained that human cooperation enables them to discuss with friends about games they would otherwise be unable to access. Without human cooperation, **P3<sub>P</sub>** would be limited to watching others play, which could lead to feelings of exclusion. According to **P4<sub>C</sub>**, human cooperation can also be used to introduce inexperienced players to gaming. For example, allowing a parent to learn to play alongside their children, thus fostering family bonding.

**Partial automation.** In the absence of a human copilot, the social dimension of partial automation may be reduced. For this reason, **P3<sub>P</sub>** would still prefer to play with **P4<sub>C</sub>** via human cooperation, even if a partial automation system were available, as those moments provide an opportunity to spend time together. **P7<sub>PC</sub>** and **P12<sub>CE</sub>** agree on this aspect, expressing that they would not want to abandon human cooperation entirely to avoid losing the social benefits it entails. In contrast, **P1<sub>P</sub>**, **P11<sub>CE</sub>**, and **P14<sub>E</sub>** observed that partial automation does not necessarily remove the social dimension; rather, it may create new opportunities. Indeed, partial automation could enable players to independently access multiplayer games, which by their nature foster social interaction and group play. Similar to human cooperation, partial automation could also facilitate the inclusion of new players in gaming.

**4.1.3 Ethical Concerns.** Some accessibility solutions can be misused or exploited for cheating, undermining the intended gameplay and negatively affecting other players' experiences. For this reason, multiplayer games may restrict their use.

**Human cooperation.** **P4<sub>C</sub>**, **P5<sub>CE</sub>**, and **P14<sub>E</sub>** note that human cooperation systems are subject to this issue. As **P2<sub>P</sub>** points out, the problem is exacerbated by the fact that multiplayer games are generally less accessible than their single-player counterparts and would therefore need better accessibility support.

**Partial automation.** A partial automation system would face similar limitations. For this reason, **P6<sub>E</sub>** is skeptical about its applicability in multiplayer contexts. **P2<sub>P</sub>** suggests distinguishing between competitive and non-competitive multiplayer games: while in the former, the use of partial automation should be regulated, in the latter it should remain unrestricted, as should be the case for single-player games. To address this concern, **P12<sub>CE</sub>** proposes the development of partial automation systems capable of learning and adapting to the player's abilities, thereby moderating the level of assistance provided. This aspect is discussed in more detail in Section 4.3.3. Overall, **P5<sub>CE</sub>** expects mixed reactions:

**P5<sub>CE</sub>:** *The people who need it would celebrate, the people who don't need it would think it's cheating. I mean, some of them, not all of them. I am able to play, but I would celebrate because I know that so many of the people that I've worked with will now be able to play more.*

## 4.2 Human Cooperation Limitations Addressed in Partial Automation

Human cooperation also has limitations that affect its widespread adoption: reduced pilot's autonomy (Section 4.2.1), need for a copilot (Section 4.2.2), and concerns about the copilot's engagement during play (Section 4.2.3).

### 4.2.1 Loss of Autonomy.

**Human cooperation.** Human cooperation implies dependence on another person, which limits the player's autonomy. Participants react to this limitation in different ways. For **P2<sub>P</sub>**, who is blind, no other accessibility solution is viable. While not uncomfortable requesting help, **P2<sub>P</sub>** acknowledges that dependence can limit play options. In contrast, **P1<sub>P</sub>** avoids human cooperation despite other tools being less effective. This is in part due to logistical barriers, and in part because having to ask for help every time would feel burdensome. **P13<sub>E</sub>** confirms this reluctance, noting that it is particularly common among people with acquired disabilities, possibly due to the inability to regain pre-disability skill levels.

**P13<sub>E</sub>:** *There will always be people who, if they can't have full access to the game they want to be able to play, would choose not to play. We have definitely met people that would say: "If I can't do this myself completely independently, I'd rather not do it." [...] I think most people would want to be able to play the way that they played before, and so it might not be as enjoyable if they don't have full access.*

For some, relying on other people's assistance should never be an option. **P2<sub>P</sub>** reports getting criticized by other players with visual impairments for relying on assistance from another person, believing that this misrepresents the gaming experience of a visually impaired player. In general, **P5<sub>CE</sub>**, **P6<sub>E</sub>**, **P13<sub>E</sub>**, and **P14<sub>E</sub>** would prioritize finding game configurations that preserve independence, without the necessity to rely on human cooperation.

**Partial automation.** Partial automation eliminates the dependency on another person, offering greater autonomy to the player. For example, if such a solution were available, **P2<sub>P</sub>** would attempt higher difficulty levels, replay games multiple times, and commit to longer sessions without the feeling of weighing on someone else. For this reason, **P6<sub>E</sub>** explained that, if partial automation were available, it would be a *second-line option*, adopted when no complete configuration based on traditional accessibility solutions can be identified. Human cooperation would therefore be considered only a *third-line option*, used solely in the absence of partial automation. Similarly, **P8<sub>PC</sub>** and **P9<sub>PC</sub>** highlighted that the independence afforded by partial automation constitutes a clear advantage over human cooperation approaches.

### 4.2.2 Copilot Availability.

**Human cooperation.** Human cooperation is also limited by the copilot's availability. One side of the problem is logistical, since the pilot and the copilot need to be physically co-located to play. All participants reported that this requirement is a significant constraint in the use of human cooperation, as it drastically reduces the pool of potential copilots. A common choice is to have a family member as the copilot, which is what **P3<sub>P</sub>** and **P4<sub>C</sub>** do. The relationship between the pilot and the copilot is discussed in Section 4.6.2. Additionally, the copilot may not always have time to play (**P3<sub>P</sub>**). To overcome the challenges of copilot's availability, some participants arrange gaming sessions in shared physical spaces. For **P7<sub>PC</sub>**, **P8<sub>PC</sub>**, **P9<sub>PC</sub>**, and **P10<sub>PC</sub>**, who meet at weekly gaming workshops, finding a copilot is straightforward, but only during scheduled sessions. An alternative, known only to **P2<sub>P</sub>**, is remote human cooperation (Section 2.3.1). However, this approach requires a complex configuration of third-party hardware and software, and it introduces interaction latency that complicates the coordination. **P2<sub>P</sub>** notes that such a solution, if natively integrated in gaming

platforms, would make it easy to meet copilots, thus improving game accessibility for many people with disabilities. When no copilot is available, fallback strategies vary. **P2<sub>P</sub>**, **P8<sub>PC</sub>**, and **P10<sub>PC</sub>** stop playing altogether; **P3<sub>P</sub>**, **P7<sub>PC</sub>**, and **P9<sub>PC</sub>** switch to more accessible alternatives, such as mobile games. Finally, **P2<sub>P</sub>**, **P9<sub>PC</sub>**, and **P10<sub>PC</sub>** sometimes play with different copilots, which introduces the challenge of learning to play together (Section 4.6.2).

*Partial automation.* Partial automation doesn't have copilot availability issues. That's why **P3<sub>P</sub>** would find a system based on it "fantastic" as it would allow playing even when no one is available as a copilot. For **P2<sub>P</sub>**, this approach would benefit the copilots as well, by freeing them from the time commitment of learning and playing the entire game.

*P2<sub>P</sub>: The benefit of it is that you don't have to then wait for a person to be available, you can play in theory any game at any time [...] you could play on your own terms without having to then rope a second person in and make them take hundreds of hours of their life.*

#### 4.2.3 Copilot's Engagement During Play.

*Human cooperation.* **P1<sub>P</sub>** notes that the gaming experience in human cooperation is not fully comparable to that offered by collaborative multiplayer games. In the latter, the roles of the two players are generally equivalent or equally significant, whereas in human cooperation, the division of tasks is often unbalanced. According to **P1<sub>P</sub>**, this imbalance can affect the copilot's enjoyment, as they are frequently assigned only a supporting role. Indeed, **P1<sub>P</sub>** considers playing as a copilot to be a predominantly negative experience, describing it as an activity performed mainly to help someone else rather than for personal enjoyment. It should be noted, however, that **P1<sub>P</sub>** has always acted as a pilot and has never directly experienced the role of copilot. In contrast, **P4<sub>C</sub>** reported finding their experience as a copilot rewarding, viewing it also as an opportunity to discover new video games.

*Partial automation.* With partial automation, this issue does not arise, as the copilot's role is replaced by a software.

### 4.3 Copilot's Interventions

The copilot can support the pilot during gameplay in various ways: assistance with the setup before starting to play (Section 4.3.1), support with menu access (Section 4.3.2), direct control of commands during gameplay (Section 4.3.3), signaling elements of interest in the game (Section 4.3.4), and suggestions on how to proceed in the game (Section 4.3.5).

*4.3.1 Assistance During Game Setup.* Setting up a game configuration is a long and complex process, which in some cases can lead to frustration (**P13<sub>E</sub>**) or even to giving up playing (**P11<sub>CE</sub>**), as also noted in prior literature [39]. The presence of a copilot can facilitate these operations.

*Human cooperation.* Some participants are only able to play if another person helps them to prepare the gaming hardware and software setup (**P3<sub>P</sub>**, **P8<sub>PC</sub>**, **P9<sub>PC</sub>**, **P10<sub>PC</sub>**, **P13<sub>E</sub>**, and **P14<sub>E</sub>**). Support can sometimes be needed even for those who can afterward play independently (**P5<sub>CE</sub>**, **P6<sub>E</sub>**, **P11<sub>CE</sub>**). For example, the player may be able to play independently using an accessible controller, but may need support to connect, position, and configure the controller itself. **P5<sub>CE</sub>** and **P6<sub>E</sub>** always try to maximize the user's autonomy, but some users still request their help in these situations.

*Partial automation.* While none of the participants commented about the (lack of) assistance during setup in a partial automation solution, we note that a software copilot cannot physically assist the pilot in preparing the gaming peripherals' configuration before starting to play. Thus, in this case, human support may still be required.

**4.3.2 Assistance with Menu Access.** Enabling in-game accessibility options through the game menu is sometimes needed to make a game accessible (**P2<sub>P</sub>**, **P5<sub>CE</sub>**, **P6<sub>E</sub>**, **P11<sub>CE</sub>**, **P12<sub>CE</sub>**, **P13<sub>E</sub>**). However, the menus themselves may lack accessibility (**P2<sub>P</sub>**, **P11<sub>CE</sub>**, **P12<sub>CE</sub>**), and solutions used to make the active gameplay accessible often do not work for accessing menus (**P12<sub>CE</sub>**). To access text from game menus and navigate them **P2<sub>P</sub>** uses Optical Character Recognition tools. This introduces additional workload and, as **P2<sub>P</sub>** points out, these technologies often make mistakes.

*Human cooperation.* Due to the above problem, **P2<sub>P</sub>** highlights that having a copilot ready to help with menu access is much preferable. **P11<sub>CE</sub>** and **P12<sub>CE</sub>** also point out the issue of menu accessibility, explaining that people participating in their workshop very rarely manage to start a gaming session without help from a copilot.

**P12<sub>CE</sub>**: *If games were designed with accessibility in mind from the start, with continuous scrolling menus where you only need to click on the options, that would be one thing. But menu accessibility is rarely implemented, even in accessible games. It may seem trivial, but it's extremely limiting for our user base.*

*Partial automation.* **P11<sub>CE</sub>** and **P12<sub>CE</sub>** highlight that partial automation could support menu navigation. By doing so, the system would allow the pilot to start a game by themselves, removing the need for other accessibility tools.

#### **4.3.3 Assistance by Playing.**

*Human cooperation.* Assistance by controlling some of the game actions is usually the main form of support provided by the copilot. **P1<sub>P</sub>**, **P3<sub>P</sub>**, **P7<sub>PC</sub>**, and **P8<sub>PC</sub>** all play with a copilot who primarily assists them by controlling actions in the game.

*Partial automation.* All participants envisioned partial automation assisting them by directly controlling some of the game actions, similarly to a human copilot. However, some expressed doubts about the copilot's ability to balance the level of support provided, avoiding being too skilled compared to the pilot and thus undermining the gameplay experience (**P1<sub>P</sub>**, **P5<sub>CE</sub>**, **P11<sub>CE</sub>**, and **P12<sub>CE</sub>**). In this sense, **P11<sub>CE</sub>** and **P12<sub>CE</sub>** expect partial automation to bring the player to "the same level as others", prioritizing the pilot's enjoyment over in-game success. According to **P5<sub>CE</sub>**, the main goal should be to enable the pilot to reach a minimum standard that ensures enjoyment, without completely replacing their skill. **P11<sub>CE</sub>** and **P12<sub>CE</sub>** also suggested that the level of support could be dynamically adjusted, progressively decreasing as the pilot improves their performance. If this is not possible, **P4<sub>C</sub>** and **P11<sub>CE</sub>** would at least like to manually select the desired level of support.

#### **4.3.4 Assistance by Signaling.**

*Human cooperation.* Another way of assisting the pilot is by signaling what's on the screen and what is happening in the game. This support is particularly valuable to **P2<sub>P</sub>**, who is blind. Indeed, **P2<sub>P</sub>** reports that their copilot must clearly indicate what is happening and which actions they are performing, so that they can coordinate effectively.

*Partial automation.* **P2<sub>P</sub>** states that partial automation should also provide this type of support, describing what is happening in the game rather than giving instructions on what to do. This way, the player can make their own decisions on how to act.

**P2<sub>P</sub>**: *If you give me that information, I can do what most players would be able to do. I'm going to decide who I target first. I'm going to decide how I play this. Whereas if you're just controlling me, I'm like "Yeah, I can shoot when you tell me to shoot, but where's the fun in that?"*

**4.3.5 Assistance by Suggesting.** Finally, the copilot may support the pilot by providing suggestions. This is particularly useful when the pilot is struggling to make progress in the game.

*Human cooperation.* **P11<sub>CE</sub>** and **P12<sub>CE</sub>** provide this type of support in their weekly gaming group. However, both note that, when giving suggestions, they always try to guide the pilot towards reasoning, without fully solving their problem.

**P11<sub>CE</sub>:** *I tell them “Let’s go back, maybe we missed something” and when we go back they notice something on their own that they didn’t see earlier, like a handle or something to pull. Then they can go on by themselves.*

*Partial automation.* In partial automation **P2<sub>P</sub>** would like a similar support, where the copilot does not intervene in the game but provides suggestions. Indeed, while it is possible for the copilot taking control of the game to resolve a situation unclear to the pilot, **P2<sub>P</sub>** notes that it is preferable not to intervene directly to preserve the pilot’s sense of agency. **P11<sub>CE</sub>** also imagines that the system could detect when the player is stuck in the game and give them suggestions to help them understand how to proceed. For **P9<sub>PC</sub>** and **P11<sub>CE</sub>**, it is important that suggestions do not directly reveal the solution, so as to leave the player with the satisfaction of solving the puzzle themselves. Finally, some participants suggest that the copilot could initiate communication proactively, for example, by recognizing moments of difficulty and spontaneously offering their help (**P1<sub>P</sub>**, **P2<sub>P</sub>**, **P12<sub>CE</sub>**).

## 4.4 Negotiating Collaboration

Participants discussed the distribution of the game actions between the pilot and the copilot (Section 4.4.1), the criteria for the action assignment (Section 4.4.2), and leadership in the decision-making. For this last aspect, we distinguish between strategic decisions, which relate to long-term goals and choices (Section 4.4.3), and tactical decisions and actions needed to address any kind of imminent situation (Section 4.4.4).

**4.4.1 Actions Separation.** The division of actions between the pilot and the copilot can be static, without changes during a game, or dynamic. It can be decided before starting, or during the game itself. Furthermore, the actions can be separated strictly, with each player taking control of an exclusive subset of actions, or overlapping, with some actions controlled by both.

*Human cooperation.* In general, **P1<sub>P</sub>** notes that the division varies depending on the game. However, **P3<sub>P</sub>** specifies that, when playing games belonging to the same genre, they often use similar configurations. The division of actions is often defined before starting to play. For example, **P1<sub>P</sub>** discusses the division with their copilot, since, having prior shared gaming experience, the copilot understands well what **P1<sub>P</sub>** is better at. Only **P2<sub>P</sub>** starts playing without a clear division of actions, preferring to work out together with the copilot what each of them does best as they play. Usually, the action separation is static throughout the game. **P2<sub>P</sub>**, instead, adopts a dynamic separation during the game: in specific sections that they would not be able to manage alone, the pilot hands over full control of the game to the copilot. Most participants adopt a strict separation between the controls managed by the pilot and those managed by the copilot. For example, **P1<sub>P</sub>**, **P3<sub>P</sub>**, and **P4<sub>C</sub>** generally choose to delegate full camera control to the copilot.

**P4<sub>C</sub>:** *One of P3<sub>P</sub>’s biggest weaknesses actually is camera control on the right stick: using the camera while doing everything else is very, very hard. So, I’d say the thing I do most is camera control. [...] I’ll point at an enemy and P3<sub>P</sub> will go ahead and handle the job.*

While strict separation of controls is more common, overlap between the pilot's and the copilot's controls is also possible. This occurs mostly in family contexts or among children, where it becomes unclear who does what (**P6<sub>E</sub>**). However, as **P13<sub>E</sub>** points out, the lack of clarity on who is responsible for which controls can lead to confusion and frustration.

*Partial automation.* In partial automation, participants also expect the ability to customize which actions to automate and how they want to be supported. As in human cooperation, **P3<sub>P</sub>**, **P4<sub>C</sub>**, **P5<sub>CE</sub>**, and **P6<sub>E</sub>** would prefer to divide the actions between pilot and copilot before starting to play. **P3<sub>P</sub>** assumes that the configuration would be similar to the one they already use when playing with a human copilot, delegating camera control to the software. Instead, **P1<sub>P</sub>** would experiment with different configurations to find the one that best suits their needs. **P3<sub>P</sub>** also hypothesizes the possibility of a dynamic division, as they would need the copilot's assistance only in specific sections of the game. Finally, **P5<sub>CE</sub>** suggests asking the software copilot to propose how to divide the actions. **P1<sub>P</sub>** expresses a similar view but emphasizes that the copilot should not be overly intrusive in suggesting configuration changes or disrupting the flow of the game.

***P5<sub>CE</sub>**: One of the things that is gonna be very important is that you understand the player, the person that you're building it for. [...] You might try to find some way to do an evaluation on the person [...], and then the AI could read what the evaluation says and say "I think I already know what you need. You need the right trigger, the right bumper acceleration, and a boost".*

#### 4.4.2 Policies Guiding Action Assignment.

*Human cooperation.* We identified three main policies guiding the allocation of controls between pilot and copilot. First, the pilot's abilities are taken into account (**P1<sub>P</sub>**, **P7<sub>PC</sub>**, **P8<sub>PC</sub>**, **P9<sub>PC</sub>**, and **P10<sub>PC</sub>**), assigning to the copilot those controls that are less accessible to the pilot. For example, **P1<sub>P</sub>** has difficulties in controlling the left hand, and therefore remaps most frequently used game actions to buttons on the right side of the controller. The remaining actions are then assigned to the copilot. When both the pilot and the copilot have disabilities, the allocation is based on the abilities of both. For instance, **P7<sub>PC</sub>**, **P8<sub>PC</sub>**, **P9<sub>PC</sub>**, and **P10<sub>PC</sub>** all have a disability and regularly play in pairs using human cooperation, supporting each other during the game. Similarly, **P5<sub>CE</sub>** reports having applied human cooperation multiple times with two people who both had disabilities. Second, as hinted above by **P1<sub>P</sub>**, the most important game actions are assigned to the pilot to ensure they maintain a sense of control over the game. For example, **P3<sub>P</sub>**, as the pilot, controls most commands, while **P4<sub>C</sub>** assists with camera control, which **P3<sub>P</sub>** cannot manage simultaneously with other actions. Third, participants divide actions based on the macro-functionality they control. For example, **P7<sub>PC</sub>** and **P8<sub>PC</sub>** separate movement from other actions, assigning all movement-related actions to one player and the remaining actions to the other.

*Partial automation.* The first two policies were also mentioned in relation to partial automation. First, participants suggested delegating to the software copilot those actions that the pilot cannot control or would struggle to control. For example, **P3<sub>P</sub>** noted that partial automation could handle complex input sequences or those requiring quick execution. Similarly, actions requiring high precision, such as aiming (**P3<sub>P</sub>**, **P9<sub>PC</sub>**), could also be delegated to the copilot. In this context, camera orientation was mentioned by many participants as a challenging action to manage and one that could be delegated to the copilot (**P1<sub>P</sub>**, **P3<sub>P</sub>**, **P7<sub>PC</sub>**). Second, participants suggested delegating to the copilot secondary actions, in particular those used only occasionally (**P1<sub>P</sub>**, **P3<sub>P</sub>**), leaving the main controls to the pilot.

#### 4.4.3 Leadership Management.

**Human cooperation.** In general, the person with disabilities acts as the pilot and makes game-wide strategic choices (**P2<sub>P</sub>**). However, when both players have a disability, the definition of who is the pilot and who is the copilot becomes more blurred. For example, **P7<sub>PC</sub>**, **P8<sub>PC</sub>**, **P9<sub>PC</sub>**, and **P10<sub>PC</sub>** all have disabilities and play in pairs through human cooperation (Section 4.4.2). Every decision is therefore made collaboratively, and both players contribute equally to the game. Finally, **P5<sub>CE</sub>** notes that sometimes a person without disabilities can be the pilot, with a person with disabilities as the copilot.

**P5<sub>CE</sub>:** *Another patient that I had who also had hemiplegia [...] and he wanted to play Call of Duty [2] with his son. And so [...] his son would look around and aim, and the father would just shoot.*

Leadership can also be temporarily assigned to the copilot. **P2<sub>P</sub>** does this in game sections that require complex or rapid input sequences. **P5<sub>CE</sub>** notes that when the copilot takes the lead, they must have experience in the specific game (Section 4.6.1), otherwise the experience could be frustrating for the pilot.

**Partial automation.** In partial automation, none of the participants mentioned the possibility of sharing or leaving game-wide strategic leadership to the software copilot. Indeed, they believe that the pilot should maintain the leadership and partial automation should intervene only when the pilot is in difficulty, and generally avoid compromising their sense of control (**P1<sub>P</sub>**, **P3<sub>P</sub>**, **P6<sub>E</sub>**). **P3<sub>P</sub>** envisions leaving temporary control to the software copilot in specific contexts, as **P2<sub>P</sub>** does in human cooperation, for example, when the game requires action execution speed that is too high for them.

**P3<sub>P</sub>:** *If the AI could change depending on the situation. I think that would be nice 'cause I hope I don't need the same level of help for an entire game. There are certain scenes that I need more help with than others.*

**4.4.4 Copilot's Operational Autonomy.** The copilot's level of operational autonomy exists on a spectrum. At the one extreme, the copilot can act as an actuator, following step-by-step instructions from the pilot, who indicates when to act and which buttons to press. At the other extreme, the copilot can be completely autonomous, executing actions based on a set of rules agreed upon with the pilot, without requiring specific instructions.

**Human cooperation.** In human cooperation, **P1<sub>P</sub>** is the only one who sometimes plays with a copilot acting solely as an actuator. This is necessary when the copilot is not experienced with the game (Section 4.6.1), as reported by **P13<sub>E</sub>**. Instead, when the copilot knows the video game, they are able to act autonomously. This results in the copilot providing more effective help, which causes the pilot to feel more immersed in the game (**P2<sub>P</sub>**). As a result, most participants play with a copilot that acts autonomously (**P1<sub>P</sub>**, **P3<sub>P</sub>**, **P4<sub>C</sub>**, **P5<sub>CE</sub>**, **P6<sub>E</sub>**, **P7<sub>PC</sub>**, **P8<sub>PC</sub>**, **P9<sub>PC</sub>**, and **P10<sub>PC</sub>**).

**Partial automation.** In partial automation, **P2<sub>P</sub>**, **P3<sub>P</sub>**, **P4<sub>C</sub>**, **P9<sub>PC</sub>**, **P11<sub>CE</sub>**, and **P12<sub>CE</sub>** envision playing with a copilot that only follows their instructions. **P9<sub>PC</sub>** specifies that they would use this support modality especially in logic games, where a more autonomous copilot might reveal the solution to proposed puzzles. **P7<sub>PC</sub>** and **P8<sub>PC</sub>**, instead, would play with a copilot that autonomously controls a predefined set of actions. **P1<sub>P</sub>** and **P5<sub>CE</sub>** also anticipate that the copilot can act autonomously, but emphasize the importance of defining rules that precisely establish what it should and can do, for example, preventing the copilot from wasting ammunition in a first-person shooter game.



## 4.5 Interaction

The interaction between pilot and copilot in shared control is fundamental for exchanging information and coordinating during gameplay. This interaction is based both on communication, which can be verbal or non-verbal (Section 4.5.1), and on the ability to understand each other's intentions (Section 4.5.2).

### 4.5.1 Verbal and Non-Verbal Communication.

*Human cooperation.* All participants interact primarily through verbal communication, engaging in dialogue with the other player to provide instructions and decide the strategy to adopt. However, the way pilots and copilots communicate often evolves over time. For example, by always collaborating with the same copilot, **P2<sub>P</sub>**, **P7<sub>PC</sub>**, and **P8<sub>PC</sub>** have developed short and quick commands over time to easily communicate with the copilot. They all believe that this way of communicating is more efficient and less tiring than using full sentences. Indeed, **P2<sub>P</sub>** recounts how their copilot acted as a guide using synthetic commands like "clear left" or "swoop" to indicate actions to perform in combat. This way, **P2<sub>P</sub>** managed to complete some particularly complex game sequences requiring direct copilot intervention only in rare cases. For **P9<sub>PC</sub>** and **P10<sub>PC</sub>**, this type of communication is particularly important in faster-paced games for real-time coordination, while in slower games the communication is more articulate and focused on making decisions jointly. Only **P7<sub>PC</sub>** also relies on non-verbal communication through eye contact, which they find more accessible due to speech difficulties.

*Partial automation.* Similarly, in partial automation, several participants propose using short and quick commands to give instructions to the copilot (**P7<sub>PC</sub>**, **P8<sub>PC</sub>**). However, **P7<sub>PC</sub>** also highlights possible difficulties related to voice use, which could slow down interaction and, for those with speech problems, be less accessible. For this reason, **P7<sub>PC</sub>** suggests teaching personalized commands to the software copilot. Alternatively, **P1<sub>P</sub>** suggests the use of brain-computer interfaces, which would enable immediate communication with the pilot, particularly suitable for fast-paced games.

### 4.5.2 Intent Understanding.

*Human cooperation.* In human cooperation, the copilot's understanding of the pilot's intentions occurs primarily through explicit requests (**P6<sub>E</sub>**). With experience, however, the copilot learns to anticipate the pilot's needs, becoming progressively more effective in providing support by observing the game and how the pilot plays (**P2<sub>P</sub>**). Similarly, the pilot's understanding of the copilot's intentions is primarily based on verbal communication (**P2<sub>P</sub>**, **P3<sub>P</sub>**, **P4<sub>C</sub>**, **P5<sub>CE</sub>**).

*Partial automation.* In partial automation, mutual understanding of intentions remains equally important (**P5<sub>CE</sub>**). If the copilot fails to understand the pilot's objective, automatic intervention might be perceived as arbitrary. **P12<sub>CE</sub>** and **P13<sub>E</sub>** agree on this issue, but also express doubts about the reliability of an automatic system in non-linear games, where possible actions are multiple and less predictable. If, instead, the pilot fails to coordinate with the copilot, they might not be able to anticipate the copilot's actions and perceive a reduced sense of control over the game (**P5<sub>CE</sub>**).

**P5<sub>CE</sub>**: *I think it would be tricky. If I had a shooting game and I wanted [the AI] to shoot for me, because I can't physically shoot the buttons. How do you control it from not using all of your ammunition? Will it just shoot all the time? Is there something that says "Ok, now I have targeted my person, and now you shoot"?*

In this context, **P12<sub>CE</sub>** proposes a dynamic interaction model, where the copilot analyzes the pilot's choices and commands, proposing interactions consistent with their intentions, and improving over time the ability to anticipate user needs.



## 4.6 Factors Affecting the Collaboration

The collaboration between pilot and copilot is influenced by several factors that determine its effectiveness. We first analyze how the copilot's knowledge of the game impacts the collaboration (Section 4.6.1). Then, we examine the impact of the relationship between the pilot and the copilot on in-game collaboration (Section 4.6.2).

### 4.6.1 Knowledge of the Game.

**Human cooperation.** Participants emphasized that the copilot should have at least a general understanding of video games and their common mechanics. **P5<sub>CE</sub>**, **P6<sub>E</sub>**, and **P11<sub>CE</sub>** consider this level of familiarity sufficient to provide effective support. **P4<sub>C</sub>** and **P6<sub>E</sub>** also mentioned specific mechanics that the copilot should be able to handle, such as camera control, character movement, and menu navigation. For this reason, **P5<sub>CE</sub>** includes a training phase for the copilot when proposing human cooperation to players with disabilities. In general, knowledge of the specific game for which support is provided is not deemed essential, but it can improve the gaming experience, particularly in complex video games (**P5<sub>CE</sub>**, **P13<sub>E</sub>**, and **P14<sub>E</sub>**). For this reason, **P4<sub>C</sub>** independently learns the game mechanics before assisting **P3<sub>P</sub>** as copilot. **P5<sub>CE</sub>** also considers experience important to facilitate communication during gameplay. For **P1<sub>P</sub>**, however, thorough knowledge of the game is essential when the copilot assumes a guiding role and needs to provide suggestions on how to proceed (Section 4.3.5). In such cases, a lack of experience with the specific game may severely undermine the effectiveness of the assistance (**P5<sub>CE</sub>**), generating frustration in the pilot, who is unable to progress further. **P2<sub>P</sub>** needs the copilot to describe what is happening on screen and therefore highlights that an experienced copilot better understands the state of the game and consequently communicates it more effectively.

**P2<sub>P</sub>**: Let's say you get a person who knows a game well and one who doesn't know it as well. Like, one knows the enemy types and the other doesn't. [...] If my copilot's saying "Okay, there's a terminator<sup>4</sup> in the distance", I'm like "Okay, I know what to do" [...] Whereas if the person doesn't know the enemies as well you then have a sense of "Oh God, this is terrifying. I don't know what's happening".

**Partial automation.** **P1<sub>P</sub>** argues that a software copilot should be trained for each video game specifically to provide effective support. However, **P1<sub>P</sub>** also questions if developing a copilot capable of adapting to different games is feasible.

### 4.6.2 Relationship Between Pilot and Copilot.

**Human cooperation.** Participants reported that the copilot is often a family member or a friend. For example, **P3<sub>P</sub>** is a family member and caregiver of **P4<sub>C</sub>**. Due to this, **P3<sub>P</sub>** is available to play together most of the time, and dedicates significant effort to better support **P4<sub>C</sub>** (e.g., by trying a game in advance before playing together). Furthermore, this type of relationship can facilitate communication, reduce logistical difficulties (Section 4.2.2), and improve coordination during play. For **P1<sub>P</sub>**, **P7<sub>PC</sub>**, **P8<sub>PC</sub>**, **P9<sub>PC</sub>**, and **P10<sub>PC</sub>** a friend acts as copilot. **P1<sub>P</sub>** highlights that prior gaming experience shared together facilitates collaboration. For these reasons, experts (**P5<sub>CE</sub>**, **P6<sub>E</sub>**, **P13<sub>E</sub>**, and **P14<sub>E</sub>**), when recommending the use of human cooperation, suggest involving someone familiar to the pilot. Familiarity is particularly important in fast-paced and complex video games (**P3<sub>P</sub>**, **P7<sub>PC</sub>**, **P8<sub>PC</sub>**), whereas it is less critical in simpler games (**P9<sub>PC</sub>**, **P10<sub>PC</sub>**). Furthermore, **P6<sub>E</sub>** notes that some of the people they assist are reluctant to accept help from someone who doesn't already support them in daily life. A copilot can also be a stranger met online. This option was mentioned by **P2<sub>P</sub>**, who also observed that this type of copilot selection can lead to variable outcomes, depending on the compatibility with the copilot.

<sup>4</sup>Terminator is a type of enemy in the video game *Warhammer 40,000: Space Marine 2*

**P2<sub>P</sub>**: *There's this idea of being drift compatible<sup>5</sup> where you are in this flow state, you are both sort of synced up with each other, and it's all about shared communication.*

**P2<sub>P</sub>** further notes that each new collaboration requires time to build the coordination needed for a smooth and satisfying gameplay experience. Indeed, the first hours of play with a new copilot are dedicated to deciding how to play together (Section 4.4), communicating (Section 4.5), and building a relationship. During this process, errors and incomprehension are possible. Thus, **P2<sub>P</sub>** and **P5<sub>CE</sub>** highlight that being positive and tolerant is important to avoid frustration.

*Partial automation.* As **P1<sub>P</sub>** pointed out, the system should adapt its assistance to the player's skill level. Otherwise, it can make the player dependent on the system, hindering autonomous growth and limiting their ability to experiment and autonomously find solutions. Similarly, **P11<sub>CE</sub>**, and **P12<sub>CE</sub>** believe that the copilot should learn the pilot's play style over time and adapt to it, as generic support may not be adequate or effective.

## 5 Discussion

Prior works highlight that shared control can be used to access games that cannot be made accessible through existing accessibility solutions [12, 25]. Our research confirms this (Section 4.1) and also uncovers *which* gaming accessibility challenges are addressed through human cooperation, whether they can be addressed through partial automation as well, and how such support should be provided (Section 5.1). The resulting design recommendations are summarized in Table 3. In particular, we discuss advantages and disadvantages of partial automation with respect to human cooperation (Section 5.2), focusing on the main limitation of human cooperation: the need for a human copilot. This necessity limits the autonomy of players with disabilities, motivating partial automation as a substitute for human cooperation. Then, based on how the pilot and the copilot collaborate in human cooperation, we discuss how this collaboration should translate into partial automation (Section 5.3). Finally, we analyze how the pilot and the copilot communicate during human cooperation, identifying possible feedback mechanisms and interactions that should be implemented in partial automation (Section 5.4).

### 5.1 Gaming Accessibility Challenges Addressed by Shared Control

The main challenge in controlling video games reported by our participants was when too many input dimensions needed to be controlled concurrently or in rapid succession. Borrowing a term from the machine learning domain [32], we call this issue the *Curse of Dimensionality* (Section 5.1.1). Unexpectedly, not all challenges addressed in shared control directly relate to control (*i.e.*, pressing buttons on the controller). Indeed, two additional challenges were also identified: the need for additional information on the game, what happens in it, and how to play it (Section 5.1.2), and the difficulty in configuring the game setup before even playing the game (Section 5.1.3).

**5.1.1 Curse of Dimensionality.** Prior works motivate the use of partial automation for game accessibility, arguing that: "Games may require inputs that some players cannot provide with any device." [12]. Our analysis confirms this, since participants reported difficulties with **fine control of input intensity**, for example, when aiming in first-person shooters or steering in driving games (Section 4.4.2). However, using existing accessibility tools and control remapping, most participants are able to mitigate this problem and can actually activate any single control individually. Instead, our analysis uncovers the curse of dimensionality problem: the participants find it difficult to **control all actions concurrently** or in rapid succession. Indeed, multiple participants reported difficulties when **simultaneously**

<sup>5</sup>*Drift compatible*: Term from the film Pacific Rim [19] in which two pilots jointly control a robot and need to be compatible to perform effectively

Table 3. Overview of partial automation design guidelines

<b>Support in controlling game actions</b>	with fine control of input intensity with concurrent multi-dimensional controls with fast-paced control sequences
<b>Support by providing information</b>	allow on-demand requests for information detect when the player does not know how to progress provide hints without spoiling the fun of the player signal elements of interest in the game
<b>Support during game setup</b>	menu navigation support through voice commands facilities for explicit and personalized control assignment propose command split between pilot and copilot
<b>Support autonomy</b>	support for different platforms and games and be tuned on them reliable level of gaming experience at all times
<b>Support sociality</b>	also support human cooperation (possibly remotely) enable multiplayer assistance
<b>Subdivision of controls criteria</b>	agency maximization considering inputs the user can control clarity of actions separation: strict and by macro-functionality
<b>Copilot's operational autonomy</b>	allow on-demand requests by the player clearly defined autonomous interventions by the copilot learn when to assist, avoid doing too much and too well baseline multiplayer assistance / adaptive to player abilities
<b>Communication from the pilot</b>	natural language requests for explicit actions or suggestions personalized shorthand commands for fast-paced gaming non-verbal commands for people with speech impairment
<b>Feedback from the copilot</b>	verbal communication to signal/suggest something non-verbal feedback through screen, sounds, light, vibrations
<b>Mutual intent understanding</b>	pilot's intent: analysis of gameplay and external sensing copilot intent: feedback through visual, auditory, haptic cues

**maneuvering multiple directional controls**, for example, the movement and the camera (Section 4.4.1). The curse of dimensionality problem is further exacerbated when a large number of controls need to be managed during **fast-paced interaction sequences** (Section 4.4.3). Human cooperation is an effective solution to address the curse of dimensionality problem since it reduces the number of controls assigned to the pilot by delegating some to the copilot. For the same reason, participants noted that partial automation could be similarly effective, highlighting its potential to replace human cooperation.

**5.1.2 Lack of Information.** In human cooperation, the copilot assists the pilot also by providing information throughout the game. Partial automation can be used to replicate such support as well (Section 4.3.5). Microsoft is currently investigating a similar type of support as part of their *Copilot* conversational assistant [42], allowing players to explicitly ask suggestions about the game they are playing. **Providing suggestions on-demand** is one way of providing information support, but partial automation should also be able to **detect when the user needs support** (e.g., does not know how to proceed) and **proactively provide suggestions**. Such support can be useful to players with various needs, including individuals with cognitive impairments or beginners. However, these **suggestions should be limited to what is needed** to progress, **without spoiling the game** for the pilot (Section 4.6.1). A special type of suggestion, particularly relevant for accessibility, is **signaling important game elements** that the player may not have perceived

(Section 4.3.4). This is highlighted by **P2<sub>P</sub>**, who is blind, but similar support could be used to signal important game sound cues to people with hearing impairments, as recently researched for real-world sound cues [29]. This kind of support is reminiscent of Navi<sup>6</sup>, the fairy companion of the protagonist in *Zelda, Ocarina of Time* [47], who alerts the player of dangers and elements of interest in the game.

**5.1.3 Support with the Game Setup.** A human copilot also provides support during the game setup. One type of support is the assistance in configuring the game setup, which can involve various hardware and software accommodations required to play a given game (Section 4.3.1). This type of support may be needed even if the game itself can be played independently. Clearly, partial automation cannot support the physical setup, which will still require human support. Instead, it can provide support during software setup, particularly in deciding how to assign commands to the pilot and the copilot. Indeed, while some participants expect the possibility to **decide control assignment explicitly**, others also suggested that partial automation could **propose how to subdivide the controls** (Section 4.4.1). This is beneficial when the pilot has no prior knowledge of the game and, therefore, has no clue on how to divide the controls. The software copilot can propose configurations based on the pilot’s disability and previous gaming sessions (Section 4.4.1).

The second type of support is for navigating the game menu, which is a common accessibility problem reported by the participants (Section 4.3.2). A partial automation system could **make the game menu more accessible**, acting as an actuator by **following the user’s explicit instructions** (Section 4.4.4).

## 5.2 Advantages (and Disadvantages) of Partial Automation over Human Cooperation

We highlight the positive impact of partial automation on the pilot’s autonomy (Section 5.2.1), and we discuss the effects of human cooperation and partial automation on sociality, inclusivity, and engagement of the players (Section 5.2.2).

**5.2.1 Autonomy.** Participants highlighted one critical limitation of human cooperation: it requires the availability of a copilot. An additional logistical constraint is that the pilot and the copilot need to be in the same place at the same time (Section 4.2.2). Remote human cooperation is one way to mitigate the constraint of being in the same place, but it still requires the availability of a human copilot and tools that are not widely known and are difficult to set up. The copilot also needs to have adequate gaming abilities and knowledge of the game, both factors which influence the resulting gaming experience (Section 4.6.1). Furthermore, at times, pilots felt that they might ask too much of copilots, and some completely gave up playing to avoid burdening the copilot (Section 4.2.1). Another concern was the possible lack of engagement of the copilot in the game, since the agency is predominantly left to the pilot (Section 4.2.3). While this specific concern seems unfounded, as the copilots we interviewed were eager to assist their pilots, it still discourages players with disabilities from asking for assistance. All these barriers collectively impede players with disabilities who play in human cooperation from playing the games they want, when they want, and in the way they want (Section 4.2.1).

Due to these reasons, participants were enthusiastic about the prospect of a software copilot. Of course, partial automation should be **available on different platforms and for different games** to be able to provide widespread and effective support. Contrary to **P1<sub>P</sub>**’s doubts on the feasibility of adapting partial automation to different games (Section 4.6.1), recent works [54], support the possibility of such a solution. Once trained on a game, unlike human cooperation, partial automation would also **ensure a consistent level of support**, and therefore, of gaming experience.

<sup>6</sup>Originally known as “Fairy Navigation System” due to its purpose of providing guidance during the game.

*5.2.2 Sociality, Inclusivity, and Engagement.* Participants reported one limitation of the idea of substituting human cooperation with partial automation: the resulting loss of live social engagement (Section 4.1.2). This is particularly true for those who organize social moments to play together and bond. Human cooperation also gives the opportunity to play together with friends and family members. Thus, participants considered partial automation not as a replacement for human cooperation, but as another option that would allow them more freedom to play in different ways. So, participants foresee the **possibility of using either human cooperation or partial automation**, according to their preferences and the availability of a copilot. Participants also note that partial automation would open up avenues for **multiplayer gaming** as another type of social interaction.

### 5.3 Collaboration Between the Pilot and the Copilot

Section 5.1 discusses what support can be provided by shared control. Here, we discuss how such support should be provided. Specifically, we discuss how the controls should be divided between the pilot and the copilot (Section 5.3.1) and the limits to when partial automation should intervene in the game (Section 5.3.2).

*5.3.1 Subdivision of the Controls.* A central aspect of shared control is determining how to divide the actions between the pilot and the copilot. We discuss the policies that should guide the configuration of a partial automation system and that could also be implemented by a partial automation system that automatically suggests a configuration.

First, the subdivision should consider how many and **which inputs can the pilot control**. Then, among all the actions in the game, the ones that provide the most sense of agency should be preferentially assigned to the input that the pilot can control, leaving the rest to the copilot. In general, the actions that **maximize the sense of agency** are the ones that are more important for the gameplay or more frequently used (Section 4.4.2). However, defining which actions are more important for the game is subjective; therefore, the process should allow for complete personalization.

Second, the subdivision of controls should take into account the **clarity of the subdivision**, avoiding sources of automation confusion [13]. Due to this, strict separation should be preferred (Section 4.4.1). Indeed, if both the pilot and the copilot control a given action, it becomes unclear who is responsible for it, and it is harder for the player to understand the consequences of their own actions. For the same reason, the actions belonging to the **same macro-functionality** should also have the same controller (Section 4.4.2).

*5.3.2 Tuning Copilot's Operational Autonomy.* Participants were concerned that partial automation interventions could compromise the pilot's sense of control. Thus, they frequently expressed the need to limit the assistance solely to situations of necessity. Interestingly, participants noted that human copilots that are not sufficiently autonomous can only act as actuators, thus spoiling the gaming experience (Section 4.4.4). In contrast, participants reported that if the software copilot is too skilled to play the game, it can reduce the pilot's sense of agency. Thus, in some cases, they suggested limiting the software agent to responding only to the **pilot's explicit commands**. This is suggested, for example, for puzzles or logic games, where the software agent solving the problem can spoil the game for the player (Section 4.4.4). Instead, for games characterized by fast-paced interactions, it could be unfeasible for the pilot to provide specific instructions, and thus the copilot should **intervene autonomously** (Section 5.1.1). For example, if the software agent is responsible for controlling the orientation in a first-person shooter, it is impossible for the pilot to provide precise real-time commands (*e.g.*, through voice) on how to adjust the orientation. It should also be possible to combine explicit actions for which the copilot solely acts as an actuator with other actions in which the copilot is more autonomous. For example, in a first-person shooter, the user can give explicit commands to change the weapon, while the software agent can autonomously aim at enemies.

In all cases, **action assignment must be clearly defined** in advance (Section 4.4.4). Additionally, the software **copilot should not be too good**, leaving space for the player’s sense of agency, and should **adapt to the player’s changing abilities** (Section 4.3.3). Autonomous interventions should also be **limited in multiplayer games** to avoid benefiting the user unfairly (Section 4.1.3). To achieve this, the support should be set to a **baseline level of abilities** (Section 4.3.3). Alternatively, the automated controls could be configured to **match the player’s skills** (Section 4.1.3).

## 5.4 Communication During the Gameplay

In human cooperation, the communication between the pilot and the copilot is fundamental. Similarly, partial automation should also provide facilities to allow communication from the pilot (Section 5.4.1), and feedback from the software copilot (Section 5.4.2). Likewise, partial automation should also foster mutual intent understanding (Section 5.4.3).

*5.4.1 Communication From the Pilot.* To support explicit requests from the pilot to the software copilot, partial automation needs to implement appropriate communication mechanisms. The main use cases for such requests are for the pilot to request some actions to the copilot (Section 4.3.3), such as to change the weapon, or to ask the copilot for suggestions on how to proceed in the game (Section 4.3.5). Participants were interested in making such requests **verbally using natural language** (Section 4.5.1). Furthermore, they also suggested the possibility of teaching **personalized shorthand commands** to the system for a quicker and more comfortable interaction.

Some participants also highlighted the need to **communicate non-verbally**, for example, in the case of players with speech impairments. For this purpose, various alternative communication channels could be used, such as head [38] or gaze [48] gestures, brain-computer interface [46], or non-verbal mouth sounds [4]. Although not explicitly mentioned by the participants, another possibility is to assign some of the game controller’s buttons to the communication with the software agent. For example, we mentioned the possibility that the pilot verbally commands the agent to set the aim on the closest enemy on the right (Section 5.3.2); an alternative solution is to use a button instead of a vocal command. Note that this is not the same as directly controlling the aim, because in this case, the pilot only activates a single button, while the copilot handles the set of actions needed to target the enemy.

*5.4.2 Feedback From the Copilot.* The software copilot should also communicate with the pilot, possibly in **natural language**, to **signal elements of interest** in the game (Section 4.3.4) and provide suggestions (Section 4.3.5). The communication should take into account **user preferences**, such as the **user-defined shorthand terms**. For effective natural language communication, Large Language Models [44] could be considered, with the prompt context taking into account information about the game and the user. In addition to verbal communication, **non-verbal options** should be possible as well, for example, by displaying **text messages on the screen**. **Symbolic communication**, through icons, **non-verbal sounds**, **light signals**, or **haptic feedback**, is also possible and should be configurable based on user preferences. For example, the copilot could highlight elements of interest on the screen through a visual marker instead of verbally communicating the position of the element of interest. Such awareness cues during gameplay are beneficial to the trust and reliance of the pilot towards the software copilot [15].

*5.4.3 Intent Understanding.* Aside from responding to explicit requests, partial automation should also attempt to **infer the user’s intended actions** to provide more meaningful automated support. Such an inference should be made by **observing the game** and the pilot’s controls. Possibly, it could also consider **additional external cues**, like the pilot’s **gaze and voice**. However, understanding the user’s intent can be challenging, in particular when the game does not

have a clear goal (Section 4.5.2). To address this issue, participants suggest that the copilot ask the pilot for confirmation on whether their intent is interpreted correctly. Of course, this should not interrupt the flow of the game.

The system should also support the pilot to understand the copilot's actions, so that the pilot can also factor the copilot's actions into the decision-making process. To this end, the copilot should provide **operational feedback** on what actions it is performing (Section 4.3.4), possibly as **visual, haptic, or sound cues**. This type of feedback can be considered a form of explainable AI [63].

## 5.5 Limitations of the Study

In our study, we employed a convenience sampling method to recruit gamers and experts with experience in human cooperation for video game accessibility. This approach allowed us to involve 14 representative participants. However, we acknowledge that, due to the sampling approach used, there are some limitations to our research. First, most of our participants have mobility impairments, with a notable exception of **P2<sub>P</sub>** who has a visual impairment. Instead, we did not have participants with hearing or cognitive impairments. Second, all our participants come from Western countries, which may introduce a cultural bias to our results. Third, our study did not include participants who were very young or elderly. These issues limit the generalizability of our findings, requiring future investigations to confirm our results.

Additionally, while all our participants had lived experience of human cooperation gaming, our findings are all based on their recounts. A direct observational study would further help identify the factors that did not transpire solely from the interviews. Furthermore, none of the participants had previously used a partial automation solution. Thus, their opinions on partial automation are all hypothetical. A possible extension of this work could therefore be a comparative observational study of human cooperation and partial automation, with the goal of confirming our current results.

## 6 Conclusions and Future Work

This work investigates current practices of shared control in video game accessibility through interviews and focus groups with players with disabilities, their supporters, and accessibility experts. Findings indicate that shared control is a crucial means of enabling access to games that would otherwise remain unplayable, while also offering opportunities for social interaction and inclusion. At the same time, reliance on human copilots introduces significant limitations, particularly regarding autonomy, availability, and engagement. Participants expressed a strong interest in partial automation as a promising extension of shared control. Such systems could increase independence, reduce logistical barriers, and broaden access. Some concerns were raised as well, in particular about preserving player agency and maintaining the social value of cooperative play. Accordingly, this study contributes to an understanding of how shared control is used in practice and how players perceive the transition toward automated copilots. On this basis, we outline design guidelines that emphasize adaptability to individual needs and mechanisms for providing support while preserving the pilot's sense of agency.

Several directions for future research emerge from our findings. First, the investigation could be extended to a broader and more diverse population of participants, including individuals with different types of disabilities, cultural backgrounds, and age groups. Such diversity would provide a more comprehensive picture of shared control practices and the potential generalizability of partial automation solutions. Second, observational studies of real-world gaming sessions are needed to complement self-reported accounts and refine our understanding of how human cooperation unfolds in practice. Such studies could reveal additional design considerations and inform the development of automation features that more closely replicate or enhance human support. Finally, the design guidelines identified in this work pave the way for prototyping and experimentally evaluating partial automation systems. Future studies could involve

user testing of such prototypes, ideally in comparison with a human cooperation baseline, to assess their effectiveness, usability, and impact on player autonomy, agency, and enjoyment.

## References

- [1] 8BitDo. n.d.. *8BitDo Lite SE*. <https://www.8bitdo.com/lite-se>
- [2] Activision. 2003. Call of Duty. Video game series. <https://www.callofduty.com>
- [3] Juan Aguado-Delgado, José-Maria Gutierrez-Martinez, José R Hilera, Luis De-Marcos, and Salvador Otón. 2020. Accessibility in video games: a systematic review. *Universal Access in the Information Society* 19, 1 (2020), 169–193.
- [4] Dragan Ahmetovic, Daniele Riboli, Cristian Bernareggi, and Sergio Mascetti. 2021. RePlay: Touchscreen interaction substitution method for accessible gaming. In *Proceedings of the 23rd International Conference on Mobile Human-Computer Interaction*. 1–12.
- [5] Paul G Backes and Kam S Tso. 1990. UMI: An interactive supervisory and shared control system for telerobotics. In *Proceedings., IEEE International Conference on Robotics and Automation*. IEEE, 1096–1101.
- [6] Kevin Bierre, Jonathan Chetwynd, Barrie Ellis, D Michelle Hinn, Stephanie Ludi, and Thomas Westin. 2005. Game not over: Accessibility issues in video games. In *Proc. of the 3rd International Conference on Universal Access in Human-Computer Interaction*. 22–27.
- [7] Kevin Bierre, Michelle Hinn, Teresa Martin, Michael McIntosh, Tess Snider, Katie Stone, and Thomas Westin. 2004. *Accessibility in Games: Motivations and Approaches*. White paper. International Game Developers Association Game Accessibility Special Interest Group. White paper.
- [8] Mark Brown and Sky LaReil Anderson. 2021. Designing for disability: Evaluating the state of accessibility design in video games. *Games and Culture* 16, 6 (2021), 702–718.
- [9] Michelle A Brown and I Scott MacKenzie. 2013. Evaluating video game controller usability as related to user hand size. In *Proceedings of the International Conference on Multimedia and Human Computer Interaction*. 1–9.
- [10] ByoWave. n.d.. *Proteus Controller*. <https://byowave.com/products/proteus-controller>
- [11] Jared E Cechanowicz, Carl Gutwin, Scott Bateman, Regan Mandryk, and Ian Stavness. 2014. Improving player balancing in racing games. In *Proceedings of the first ACM SIGCHI annual symposium on Computer-human interaction in play*. 47–56.
- [12] Gabriele Cimolino, Sussan Askari, and TC Nicholas Graham. 2021. The role of partial automation in increasing the accessibility of digital games. *Proceedings of the ACM on Human-Computer Interaction* 5, CHI PLAY (2021), 1–30.
- [13] Gabriele Cimolino, Renee Chen, Carl Gutwin, and TC Nicholas Graham. 2023. Automation Confusion: A Grounded Theory of Non-Gamers' Confusion in Partially Automated Action Games. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–19.
- [14] Gabriele Cimolino and TC Nicholas Graham. 2022. Two heads are better than one: A dimension space for unifying human and artificial intelligence in shared control. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 1–21.
- [15] Gabriele Cimolino, Carl Gutwin, and T.C. Graham. 2022. Impact of Awareness Cues on Trust in Human-AI Shared Control. In *TRAIT: Workshop on Trust and Reliance in AI-Human Teams, at CHI 2022*. ACM.
- [16] ConsoleTuner. n.d.. *Titan Two*. <https://www.consoletuner.com/products/titan-two>
- [17] Flex Controller. n.d.. *Flex Controller*. <https://www.flex-controller.com>
- [18] Mat Dalglish. 2023. Who Can Play? Rethinking Video Game Controllers and Accessibility. In *Disability and Video Games: Practices of En-/Disabling Modes of Digital Gaming*. Springer, 43–71.
- [19] Guillermo del Toro. 2013. *Pacific Rim*. Warner Bros. Pictures and Legendary Pictures. (Motion picture).
- [20] Paul Dietz and Darren Leigh. 2001. DiamondTouch: a multi-user touch technology. In *Proceedings of the 14th annual ACM symposium on User interface software and technology*. 219–226.
- [21] Anca D Dragan and Siddhartha S Srinivasa. 2013. A policy-blending formalism for shared control. *The International Journal of Robotics Research* 32, 7 (2013), 790–805.
- [22] Nintendo EAD. 2017. *Mario Kart 8 Deluxe*. Video game, released on Nintendo Switch.
- [23] Sony Interactive Entertainment. 2023. *PS5 Access Controller*. <https://www.playstation.com/accessories/access-controller>
- [24] Daniela Feth, Binh An Tran, Raphaela Groten, Angelika Peer, and Martin Buss. 2009. Shared-control paradigms in multi-operator-single-robot teleoperation. In *Human Centered Robot Systems: Cognition, Interaction, Technology*. Springer, 53–62.
- [25] David Gonçalves, Manuel Piçarra, Pedro Pais, João Guerreiro, and André Rodrigues. 2023. "My Zelda Cane": Strategies Used by Blind Players to Play Visual-Centric Digital Games. In *Proceedings of the 2023 CHI conference on human factors in computing systems*. 1–15.
- [26] David Gonçalves, André Rodrigues, Mike L Richardson, Alexandra A De Sousa, Michael J Proulx, and Tiago Guerreiro. 2021. Exploring asymmetric roles in mixed-ability gaming. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–14.
- [27] Deepak Gopinath, Siddarth Jain, and Brenna D Argall. 2016. Human-in-the-loop optimization of shared autonomy in assistive robotics. *IEEE robotics and automation letters* 2, 1 (2016), 247–254.
- [28] Bastian Ilse Hougard, Ingeborg Goll Rossau, Jędrzej Jacek Czaplą, Mozes Adorjan Miko, Rasmus Bugge Skammelsen, Hendrik Knoche, and Mads Jochumsen. 2021. Who willed it? decreasing frustration by manipulating perceived control through fabricated input for stroke rehabilitation BCI games. *Proceedings of the ACM on Human-Computer Interaction* 5, CHI PLAY (2021), 1–19.



- [29] Jeremy Zhengqi Huang, Hriday Chhabria, and Dhruv Jain. 2023. “Not There Yet”: Feasibility and Challenges of Mobile Sound Recognition to Support Deaf and Hard-of-Hearing People. In *Proceedings of the 25th International ACM SIGACCESS Conference on Computers and Accessibility*. 1–14.
- [30] Susan Hwang, Adrian L Jessup Schneider, Daniel Clarke, Alexander Macintosh, Lauren Switzer, Darcy Fehlings, and TC Nicholas Graham. 2017. How game balancing affects play: Player adaptation in an exergame for children with cerebral palsy. In *Proceedings of the 2017 conference on designing interactive systems*. 699–710.
- [31] JoyToKey. n.d.. *JoyToKey*. <https://joytokey.net/en>
- [32] Eamonn Keogh and Abdullah Mueen. 2011. Curse of dimensionality. In *Encyclopedia of machine learning*. Springer, 257–258.
- [33] Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser. 2017. *Research methods in human-computer interaction*. Morgan Kaufmann.
- [34] Mingjun Li, Haotian Cao, Xiaolin Song, Yanjun Huang, Jianqiang Wang, and Zhi Huang. 2018. Shared control driver assistance system based on driving intention and situation assessment. *IEEE Transactions on Industrial Informatics* 14, 11 (2018), 4982–4994.
- [35] Anna Loparev, Walter S. Lasecki, Kyle I. Murray, and Jeffrey P. Bigham. 2014. Introducing shared character control to existing video games. In *International Conference on Foundations of Digital Games*.
- [36] William Lu. 2008. Evolution of video game controllers. *Roseville: Prima Publishing* 2 (2008).
- [37] Dario Maggiorini, Marco Granato, Laura Anna Ripamonti, Matteo Marras, and Davide Gadia. 2017. Evolution of game controllers: Toward the support of gamers with physical disabilities. In *International Conference on Computer-Human Interaction Research and Applications*. Springer, 66–89.
- [38] Matteo Manzoni, Dragan Ahmetovic, and Sergio Mascetti. 2024. Personalized Facial Gesture Recognition for Accessible Mobile Gaming. In *International Conference on Computers Helping People with Special Needs*. Springer, 120–127.
- [39] Jesse J Martinez, Jon E Froehlich, and James Fogarty. 2024. Playing on hard mode: Accessibility, difficulty and joy in video game adoption for gamers with disabilities. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. 1–17.
- [40] Lucas Medeiros and Flavio Coutinho. 2015. Developing an Accessible One-Switch Game for Motor Impaired Players. *Proceedings of SBGames* (2015), 236–239.
- [41] Microsoft. 2017. *Xbox Controller Assist (formerly Xbox Copilot)*. <https://support.xbox.com/en-US/help/account-profile/accessibility/copilot>
- [42] Microsoft. 2025. *Gaming Copilot*. <https://news.xbox.com/2025/08/06/gaming-copilot-beta-begins-rolling-out-to-xbox-insiders-on-game-bar-today>
- [43] Microsoft. n.d.. *Xbox Adaptive Controller*. <https://www.xbox.com/accessories/controllers/xbox-adaptive-controller>
- [44] Bonan Min, Hayley Ross, Elinor Sulem, Amir Pouran Ben Veyseh, Thien Huu Nguyen, Oscar Sainz, Eneko Agirre, Ilana Heintz, and Dan Roth. 2023. Recent advances in natural language processing via large pre-trained language models: A survey. *Computing Surveys* (2023).
- [45] Sarah Mosely, Raeda Anderson, George Usmanov, John Morris, and Ben Lippincott. 2022. Video game trends over time for people with disabilities. *The Journal on Technology and Persons with Disabilities* 232 (2022).
- [46] Tutan Nama and Debasis Samanta. 2024. QC Speller: User Interface Design of a Hands-Free Touch-Free Speller with Brain Electroencephalogram Sensory Rhythm. *Transactions on Accessible Computing* (2024).
- [47] Nintendo. 1998. The Legend of Zelda: Ocarina of Time. Video game, originally released on Nintendo 64, and later on Nintendo GameCube. <https://www.nintendo.co.jp/n01/n64/software/zelda/index.html>
- [48] Youn Soo Park and Roberto Manduchi. 2024. A Functional Usability Analysis of Appearance-Based Gaze Tracking for Accessibility. In *Symposium on Eye Tracking Research and Applications*.
- [49] PlatinumGames. 2014. Bayonetta. Video game, originally released on Xbox 360, PlayStation 3, and later on Nintendo Wii U, Nintendo Switch, Xbox One, PlayStation 4, and PC.
- [50] PlatinumGames. 2014. Bayonetta 2. Video game, originally released on Nintendo Wii U and later on Nintendo Switch.
- [51] PlayAbility. n.d.. *PlayAbility*. <https://www.playability.gg>
- [52] Priori Data. 2025. *How Many Gamers Are There in 2025? Latest Stats*. <https://prioridata.com/number-of-gamers/>
- [53] Marco C Rozendaal, Bram AL Braat, and Stephan AG Wensveen. 2010. Exploring sociality and engagement in play through game-control distribution. *Ai & Society* 25 (2010), 193–201.
- [54] SIMA Team, Maria Abi Raad, Arun Ahuja, Catarina Barros, Frederic Besse, Andrew Bolt, Adrian Bolton, Bethanie Brownfield, Gavin Buttimore, Max Cant, Sarah Chakera, Stephanie C. Y. Chan, Jeff Clune, Adrian Collister, Vikki Copeman, Alex Cullum, Ishita Dasgupta, Dario de Cesare, Julia Di Trapani, Yari Donchev, Emma Dunleavy, Martin Engelcke, Ryan Faulkner, Frankie Garcia, Charles Gbadamosi, Zhitaogong, Lucy Gonzales, Kshitij Gupta, Karol Gregor, Arne Olav Hallingstad, Tim Harley, Sam Haves, Felix Hill, Ed Hirst, Drew A. Hudson, Jony Hudson, Steph Hughes-Fitt, Danilo J. Rezende, Mimi Jasarevic, Laura Kampis, Rosemary Ke, Thomas Keck, Junkyung Kim, Oscar Knagg, Kavya Kopparapu, Rory Lawton, Andrew Lampinen, Shane Legg, Alexander Lerchner, Marjorie Limont, Yulan Liu, Maria Loks-Thompson, Joseph Marino, Kathryn Martin Cussons, Loic Matthey, Siobhan McLoughlin, Piermaria Mendolicchio, Hamza Merzic, Anna Mitenkova, Alexandre Moufaret, Valeria Oliveira, Yanko Oliveira, Hannah Openshaw, Renke Pan, Aneesh Pappu, Alex Platonov, Ollie Purkiss, David Reichert, John Reid, Pierre Harvey Richemond, Tyson Roberts, Giles Ruscoe, Jaime Sanchez Elias, Tasha Sanders, Daniel P. Sawyer, Tim Scholtes, Guy Simmons, Daniel Slater, Hubert Soyer, Heiko Strathmann, Peter Stys, Allison C. Tam, Denis Teplyashin, Tayfun Terzi, Davide Vercelli, Bojan Vujatovic, Marcus Wainwright, Jane X. Wang, Zhengdong Wang, Daan Wierstra, Duncan Williams, Nathaniel Wong, Sarah York, and Nick Young. 2024. Scaling Instructable Agents Across Many Simulated Worlds. arXiv:2404.10179 [cs.LG] <https://arxiv.org/abs/2404.10179>
- [55] Warhorse Studios. 2018. Kingdom Come: Deliverance. Video game, released on Microsoft Windows, PlayStation 4, and Xbox One.
- [56] Philipp Sykownik, Katharina Emmerich, and Maic Masuch. 2017. Exploring patterns of shared control in digital multiplayer games. In *International Conference on Advances in Computer Entertainment*. Springer, 847–867.

- [57] Gareth Terry, Nikki Hayfield, Victoria Clarke, Virginia Braun, et al. 2017. Thematic analysis. *The SAGE handbook of qualitative research in psychology* 2, 17-37 (2017), 25.
- [58] Turn 10 Studios. 2023. *Forza Motorsport Accessibility Support*. <https://support.forzamotorsport.net/hc/en-us/articles/20964254277267-Forza-Motorsport-Accessibility-Support>
- [59] Rodrigo Vicencio-Moreira, Regan L Mandryk, Carl Gutwin, and Scott Bateman. 2014. The effectiveness (or lack thereof) of aim-assist techniques in first-person shooter games. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 937–946.
- [60] VoiceAttack. n.d.. *VoiceAttack*. <https://voiceattack.com>
- [61] Dakuo Wang, Elizabeth Churchill, Pattie Maes, Xiangmin Fan, Ben Shneiderman, Yuanchun Shi, and Qianying Wang. 2020. From human-human collaboration to Human-AI collaboration: Designing AI systems that can work together with people. In *Extended abstracts of the 2020 CHI conference on human factors in computing systems*. 1–6.
- [62] Xbox. n.d.. *Xbox Adaptive Joystick*. <https://www.xbox.com/accessories/controllers/xbox-adaptive-joystick>
- [63] Feiyu Xu, Hans Uszkoreit, Yangzhou Du, Wei Fan, Dongyan Zhao, and Jun Zhu. 2019. Explainable AI: A brief survey on history, research areas, approaches and challenges. In *CCF international conference on natural language processing and Chinese computing*. Springer, 563–574.
- [64] Bei Yuan, Eelke Folmer, and Frederick C Harris Jr. 2011. Game accessibility: a survey. *Universal Access in the information Society* 10, 1 (2011), 81–100.