

Failing with Style: Designing for Aesthetic Failure in Interactive Performance

¹Adrian Hazzard, ¹Chris Greenhalgh, ²Maria Kallionpää, ¹Steve Benford,

³Anne Veinberg, ⁴Zubin Kanga & ⁵Andrew McPherson

¹Mixed Reality Lab, University of Nottingham; ²Music & Sound Knowledge Group, Aalborg University;

³Leiden University, Netherlands; ⁴Royal Holloway, University of London;

⁵Centre for Digital Music, Queen Mary University of London

¹[firstname.lastname]@nottingham.ac.uk; ²kallionpaa@hum.aau.dk;

³a.veinberg@hum.leidenuniv.nl; ⁴zubin.kanga@rhul.ac.uk; ⁵a.mcpherson@qmul.ac.uk

ABSTRACT

Failure is a common artefact of challenging experiences, a fact of life for interactive systems but also a resource for aesthetic and improvisational performance. We present a study of how three professional pianists performed an interactive piano composition that included playing hidden codes within the music so as to control their path through the piece and trigger system actions. We reveal how apparent failures to play the codes occurred for diverse reasons including mistakes in their playing, limitations of the system, but also deliberate failures as a way of controlling the system, and how these failures provoked aesthetic and improvised responses from the performers. We propose that creative and performative interfaces should be designed to enable aesthetic failures and introduce a taxonomy that compares human approaches to failure with approaches to capable systems, revealing new creative design strategies of gaming, taming, riding and serving the system.

CCS CONCEPTS

• Human-centered computing-Empirical studies in HCI • Human-centered computing-Interaction design theory, concepts and paradigms • Applied computing-Sound and music computing

KEYWORDS

Musical performance; aesthetic failure; interactive systems; design; musical codes.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.

CHI 2019, May 4–9, 2019, Glasgow, Scotland UK

© 2019 Copyright is held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-1-4503-5970-2/19/05...\$15.00

<https://doi.org/10.1145/3290605.3300260>

ACM Reference format:

Adrian Hazzard, Chris Greenhalgh, Maria Kallionpää, Steve Benford, Anne Veinberg, Zubin Kanga & Andrew McPherson. 2019. Failing with Style: Designing for Aesthetic Failure in Interactive Performance. In *2019 CHI Conference on Human Factors in Computing Systems Proceedings (CHI 2019)*, May 4–9, 2019, Glasgow, Scotland, UK. ACM, New York, NY, USA. Paper 30, 13 pages. <https://doi.org/10.1145/3290605.3300260>

1 INTRODUCTION

As computer systems become evermore capable, exceeding human capabilities in some respects and assuming human-like qualities, so the matter of failure in HCI takes on new meaning. Dependable systems engineering has long recognised the need to consider humans when accounting for and trying to prevent system failure [1]. HCI has often seen failure as a problem to be avoided, for example considering how to support humans as the ‘weakest links’ in secure authentication [38] or how elderly users may be inhibited by a fear of failure [13]. However, seen from a different perspective, failure is the inevitable consequence of striving to succeed. In interactive systems, this is most evident in the domain of games where players undertake and routinely fail at difficult interactional challenges [25]. Enjoying the risk and reality of failure is an inherent part of much play [24], while in creative performance the prospect or occurrence of failure may require performers to improvise or adopt appropriate strategies in response.

In order to unpack the complex and aesthetic nature of failure in performance we present a study of how highly proficient humans—professional concert pianists—engaged a highly capable interface—a self-playing piano—to perform an interactive and game-like classical piano composition. In this paper we focus on failure as viewed from the performer’s perspective. We show how failure can be analysed as a multi-layered phenomenon. In this particular case we identify and distinguish failures that: remain within the narrative of the work; compromise the musicality of the performance; compromise the integrity of the specific work;

and prevent any kind of performance. This allows us to see that an apparently simple user error such as playing a note not on the musical score may be part of a deliberate strategy to fail at one level (within the narrative of the work) while succeeding at another level (giving an enjoyable musical performance of the work). We enrich HCI's ability to reason about and design for failure by introducing a 'taxonomy of aesthetic failure' that compares users' approaches to failure with their approaches to the system, revealing how designers might enable humans to game, tame, ride, serve or negotiate with capable interfaces.

2 RELATED WORK

2.1 Failure in musical performance

The notion of failure is typically considered undesirable for both humans and capable systems. Within musical performance, especially in the classical music domain, failure is seen as a problem to be avoided: the performer should faithfully reproduce the intentions of the composer by strictly following the score [4]. Whereas, subtle variations from the score form the performer's expressive interpretation [15], insertions, deletions or reordering's would be considered errors and a failure to perform the work correctly. In this view, failure of interaction between humans and systems pivots on "*the disparity between the hypothetical sequence of events on which the design is based, and the action's actual course*" [42].

In other contexts, the ways in which computer systems deviate from human musicians become an explicit part of the musical style. Jean-Claude Risset's seminal piece *Duet for One Pianist* for Disklavier piano (1989) builds on its ability to play with speed, precision and spread of notes that human hands could never achieve, while working around the Disklavier's long latency in comparison to human pianists [36]. Elsewhere, entire genres of music such as *glitch* and *chiptunes* have built up around the non-idealities and artefacts of digital audio processing [8].

Relatively less has been written about the opportunities for human failure in musical performance, although Bin [4] and Newland [33] highlight the importance of the element of *risk* in producing an engaging performance. Musical expertise and virtuosity are not necessarily enough to remove human error, as [31,44] noted that well trained musicians regularly make errors but are well versed at correcting or disguising them. However, for computer score following systems such inaudible deviations pose a challenge [11]. Tom Johnson draws special attention to human limitations in his piece *Failing: a very difficult piece*

for solo string bass (1975), in which the performer attempts to read a lengthy narrative text while simultaneously playing music of increasing technical difficulty. Meanwhile the spoken narrative questions the idea of failure by asking whether the performer could "fail to fail" should they not make mistakes.

As ever, artists working at the cutting edge of performance may be anticipating issues that will eventually surface in the mainstream. As we enter a world in which AI and robotics begin to impinge on more mainstream musical activity—from 'intelligent drummers'¹ in digital audio workstations to robot jazz playing musicians on stage [20], so human performers are increasingly engaging—or are in danger of being replaced by—capable musical interfaces. In this light, it is important to consider how human failure with such technologies can become a creative and aesthetic matter and also how interfaces can be designed to fail for themselves in aesthetic ways.

2.2 Musical games and failure.

Games and gaming have a long history in music, from the dice games attributed to composers such as Mozart and C.P.E Bach in the 1700's [21], to 20th Century aleatoric compositions like *Reunion* by John Cage, where the movements of pieces in a game of Chess played on stage "would result in the selection of sound sources and their spatial distribution around the audience" [12]. Contemporary examples combine music performance with digital gaming, for instance Joost van Dongen's *Cello Fortress* [10] and Turowski & Hutchinson's *Plurality Spring* [41], where players control on-screen elements (e.g. weapons or avatars) via the performance of musical material. These examples combine gameplay and musical improvisation, granting the performer free rein and flexibility of their performance. In contrast, the commercial music game *Guitar Hero* [19] offers little room for failure or improvisation, as the game highlights deviations with unpleasant sounds, while Karaoke games such as *Sing Star* [40] permit a degree of creative flexibility in that the performer can deviate should they wish, but at a cost to their overall score.

Juul [25] points to the paradox of failure in digital computer games, asserting that humans typically avoid failure in life, but nonetheless failure (i.e., a breakdown) is often built into games, and humans seek out and enjoy playing games. Looking beyond individual player actions and their successes and failures within the game narrative

¹ https://support.apple.com/kb/PH13070?locale=en_US

layer, Ryan and Siegl [37] chart more subtle forms of failure or ‘breakdowns’ that can undermine the player’s experience of the game as whole, and patterns for avoiding them. Iacovides et al. [24] distinguish between breakdowns in usability, learning and user engagement, and stress that players need to understand the nature of the breakdowns in order to learn from them. Similarly, in HCI, error management training embraces “*active exploration as well as explicit encouragement for learners to make errors during training and to learn from them*” [28].

3 APPROACH

Recent years have seen HCI embrace cultural applications of computing which has led to the emergence of new methods that respect artistic approaches, including Performance-led Research in the Wild, which we adopt here [2]. The approach is one of learning through collaboration with artists whose vision initially drives the research agenda, which HCI researchers then help shape, implement and study. The approach can therefore be seen as a form of Research Through Design in which findings emerge from reflection on exploratory practice rather than being hypothesis driven [45]. The approach also takes place ‘in the wild’ [2] so the artist’s practice remains legitimate and the findings emerge from the rigors of—as in our case—live public performance rather than lab experiments or demonstrators.

The following draws on a detailed analysis of how the composer and two other professional concert pianists performed an interactive piano work called *Climb!* on 10 occasions at five different venues. Our goal in recruiting two other pianists—in addition to the composer—to perform the work was to obtain contrasting perspectives. The second pianist (labelled performer A) was given full knowledge of the composition’s game-like interactivity and access to the software system for rehearsal, but obviously lacked the composer’s (M) ‘inside knowledge’ of the work. The third pianist (labelled performer Z) was given only limited details of the work’s interactivity (explained below), in order to explore how a performer would approach the performance of such a work when elements of its interactivity are unknown. Although performers A and Z performed *Climb!* only once, their engagement with the work from initial contact, through rehearsal and preparation, to the moment of performance was deep, prolonged and yielded rich, and fascinating data. We begin by describing *Climb!* and the composer’s design rationale, before revealing how our pianists approached performing the work.

4 INTRODUCING CLIMB!

Climb! is a classical piano duet written for human pianist and a Disklavier (self-playing) piano. The concept of *Climb!* combines a traditional virtuoso concert work with the narrative interaction of a computer game [26,27]. A study of how audiences interpreted the work over the course of repeat performances is reported in [3].

4.1 Duetting with a Disklavier Piano

Climb! is performed with a Yamaha Disklavier piano. This is a contemporary digital version of the traditional self-playing piano. Incoming MIDI (Musical Instrument Digital Interface) messages define the notes to be played, their timing, duration and velocity (loudness). Actuators integrated into the instrument move the keys and strike the strings accordingly. At the same time, a human pianist’s performance on the piano is output as MIDI which can then be sent to other MIDI enabled devices. In *Climb!* the Disklavier is employed to perform elements of the music that are beyond the capability of even a professional concert pianist, extending the range of music that can be played and also challenging the human musician. *Climb!* involves control passing back and forth between the human and piano, with each performing solo at some points, but with many passages where the pair duet. When duetting occurs the Disklavier plays a pre-programmed sequence of notes (triggered through MIDI) that the human plays along with, interleaving their fingers with the self-actuated moving keys of the piano. Most of the sections of *Climb!* contain such duets, some with the piano playing first and others with the pianist playing first.

4.2 Navigating a Non-linear Score

Like many classical compositions, *Climb!* is a scored work. The composer has written a score that shows the parts that the human is intended to play alongside the parts the system will play in traditional western music notation. Unusually, the score is non-linear, offering the human performer a choice of routes through the overall piece. The work embraces the narrative metaphor of a challenging *Climb!* up a mountain. As can be seen in Figure 1 (which also illustrates the performances considered in this paper), this is formed around a set of 26 scored sections that represent various ‘events’ that the pianist encounters as they ascend the mountain (shown as grey rectangles). These events broadly conform to the narrative metaphor, for instance the pianist encounters hallucinations, a mysterious forest, falling rocks, animals, and changing weather conditions, all of which are represented by the music. These events are arranged into three ‘paths’ that

symbolise distinct journeys up the mountain (top, middle and bottom of Figure 1). Each performance commences at *Basecamp* and concludes at *Summit*. Musical ‘codes’ (see below) are embedded in many sections of the work; whether these codes are played successfully determines how the piece progresses. Consequently, a performance of *Climb!* does not include all 26 sections, but rather follows a weaving trajectory through a particular sequence of musical events. The musical codes give variation and flexibility to the overall form: because *Climb!* is also a game, it is not desirable for the pianist to pre-select and over-practice any particular route. Although this would be typical for a classical music pianist, it would strip *Climb!* of its originality and non-linear form, also potentially make the piece less interesting for the audience to follow.

4.3 Interacting Through Embedded Codes

The key interaction mechanism used throughout *Climb!* is to embed musical ‘codes’ as triggers within the score as described in [18]. The system listens continuously to the notes being played by the pianist (using MIDI) and matches this input stream against pre-authored candidate codes, isolated musical phrases specified as sequences of note pitches and durations. These embedded musical codes fulfil multiple purposes as explained below.

4.3.1 Challenge codes

These determine the route that a performer will take through the score. If a performer’s plays a Challenge code correctly then they are considered to have ‘won’ the musical challenge in that section and will continue up their chosen path. If the code is not matched correctly by the system, then they are forced to move to another path. There are 12 Challenge codes distributed throughout the score and they are typically the longest (i.e. most number of note events) and hardest to perform. They are shown as diamonds in Figure 1. Some of the Challenge codes are clearly notated and likely to be played correctly when the pianist has rehearsed the work. Other challenge codes are harder to perform or are notated “ambiguously”, in a manner that can encourage the performers to treat them rather freely, which may lead to the system not recognizing them (and thus interpreting them as “failed”; see vignette A). Such codes mostly appear towards the end of the work, their purpose being to provide more dramatic moments towards its culmination.

4.3.2 Choice codes

Choice codes are only encountered in *Basecamp*. They function the same as the challenge codes described above

except that the pianist is explicitly presented with a choice of 3 codes (musical phrases) to perform. If one is performed and matched correctly that determines the initial path taken. If none of the choices are played correctly then the system navigates to path 1 by default. These are also shown as diamonds in Figure 1, in *Basecamp* (only).

4.3.3 Disklavier codes

Disklavier codes trigger the piano to self-play during the duets where the human starts first. They are typically formed from short easy sequences of pitches (i.e. 1 – 4 notes), explicitly marked in the piano score, as it is vital that they are easy to play and can be reliably recognised so as to trigger the Disklavier part as the composer intended. These are shown as right-facing triangles in Figure 1. Finally, there is also one Approach code (left-facing triangle), which notifies those audience members using the *Climb!* web app [3] that a Challenge code is approaching, but this is not considered in depth in this paper.

5 CLIMB! PERFORMANCES

We were able to observe and document five concerts of *Climb!*. Breaking with typical performance tradition *Climb!* was performed twice back to back at each concert (with the exception of the second), so as to give the audience a deeper understanding and appreciation of the work’s non-linear form (see also [3]). In total we captured data from 10 performances of *Climb!* (including the dress rehearsal before the first public performance), referred to in the findings by the number ordering, ‘P1’–‘P10’. These performances were given by three professional pianists, identified here as Performer M, Performer A and Performer Z. All regularly perform contemporary classical / electro-acoustic repertoire, which often includes works that embrace novel digital technologies. Performer M, who also composed *Climb!*, played in eight of the performances. Performers A and Z played one performance each (P5 and P6 respectively, both in the third concert). To rehearse, Performer A was given full access to the score and system for *Climb!*. However, Performer Z was not made aware of the location of the Challenge codes and was not given access to the interactive system until the final rehearsals at the venue. This reflected the composer’s concern that pianists who know the location of Challenge codes might over-rehearse them, thus rendering them less challenging and diminishing the indeterminacy of the work’s form. It also provided us with an opportunity to observe how a pianist comes to terms with an interactive piece when some of the principle interactions are unknown to them.

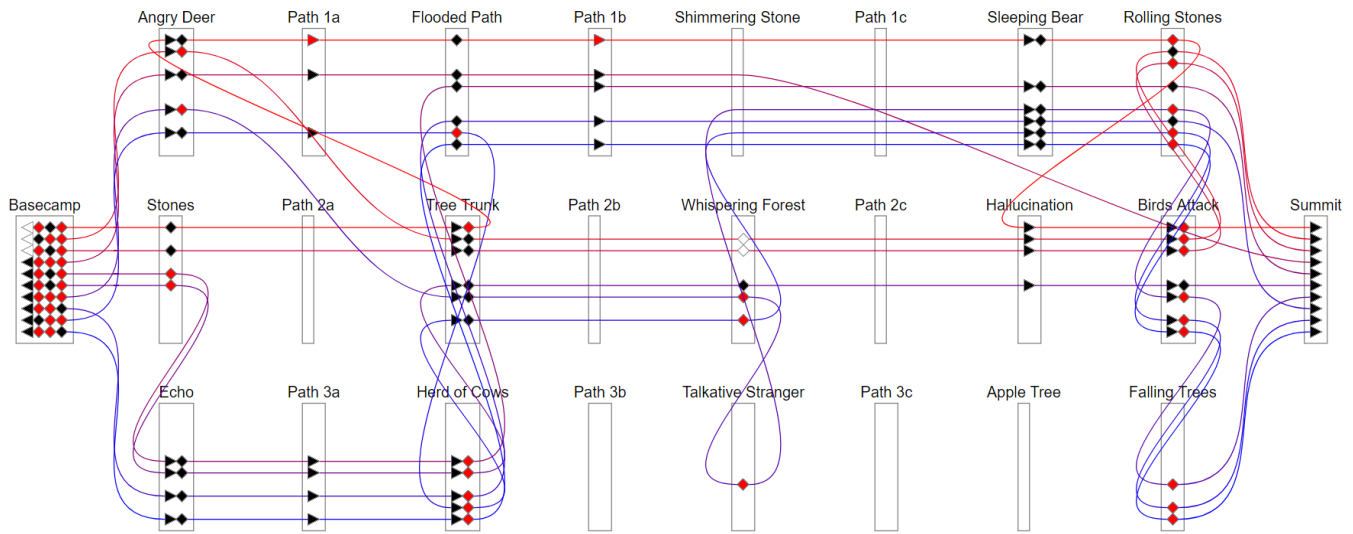


Figure 1: Map of Climb! performances

6 DATA CAPTURE AND ANALYSIS

We captured the end-to-end rehearsal and performance process, which included: records of our communications (e.g., emails) with the performers throughout their orientation and rehearsal processes; pre- and post-performance semi-structured interviews with each performer captured on video; system logs of each performance that detailed all system interactions (e.g. codes triggered), and the pianist's individual performances as MIDI files and video/audio recordings; and finally, a written reflection by each performer in the weeks following their performance. Our focus was on the performers experience rather than the audiences. Our analysis of the captured data revealed a range of types and motivations for failure which informed the emergence of our taxonomy. Preliminary analysis of interviews and writings provided an initial orientation to performers' strategies and highlighted that diverse kinds of failure were inherent to the performances. We then processed the system logs to map all occurrences of failure to complete a code (Figure 1). For each such case, we compared the MIDI file of the performance (showing notes and rhythms played) with the video and the original score to pinpoint the specific nature and cause of the failure. Finally, we turned to the interviews and performers' writings (and exchanged further emails when necessary) to confirm our reasoning and captured the performers' accounts of what had occurred, including their motivations.

In writing the paper, we selected vignettes to best illustrate the breadth of failures that occurred across all performances. After a general overview of the

performances and the strategies our performers took in approaching them, we then report 5 vignettes involving performer M, 3 from performer A, and 1 from performer Z, each exposing a specific type of failure.

7 FINDINGS: THE DIVERSITY OF FAILURE

7.1 General Orientation

Here we present an overview of the 10 performances of Climb!. Figure 1 shows the trajectory of each performance (in descending order, with P1 at the top). Shapes filled solid black indicate codes performed and matched correctly, whereas shapes filled red indicate codes that were not matched. Unfilled shapes indicate that the code was not present in the version of Climb! used for that performance. Each performance (except for P4) commenced at *Basecamp* and concluded at *Summit* and featured a full sequence of sections in-between. The second concert performance (P4) experienced major technical problems which resulted in an incomplete reading as detailed in vignettes H and I, below. This overview reveals how failure to play and/or match codes was commonplace. All performances branched between paths at least once, while P9 branched five times. Three sections in path 3 have never been performed in a concert setting. All performances contained a number of unmatched codes, ranging from 1 in P8 to 6 in P9. In total 33 codes were not successfully matched, 31 of these being Challenge codes, and the remaining two were Disklavier codes. However, while 'failure' was clearly inherent to the work, it transpires that it occurred in many ways, with different causes, intents and consequences.

7.2 Performers' Strategies

In order to understand this diversity of apparent failure we now consider the performer's strategies for the work.

Performer M, the composer of the work was for obvious reasons completely familiar with the musical material prior to the rehearsals. However, playing an entire composition in a concert situation is more demanding than testing parts of it during composition, and Performer M reported that she learned to play the work only once she started to rehearse it on the actual Disklavier that was used in the first concert: although she had written the music, the physical interaction with the new instrument had to be specifically learned and focused on. According to Performer M, in the first performance of *Climb!* she mainly concentrated on the composition being communicated effectively, and that *"everything would work"*, sometimes even at the expense of the pianistic details. She reported that in the first performance, as well as in certain later performances, interacting with the electronic system took most of her attention. However, she felt that knowing the music and the system did give her confidence to perform the piece.

The first stage of **Performer Z's** rehearsal process was to learn the score, specifically the notes and rhythms required of the human performer in isolation. This then transitioned into a period of working with provided audio files of the Disklavier part. At first he practiced with a set of audio files containing a click track (i.e., giving exact timing cues), and then moving on to a set without the click-track: *"Practicing my part alongside the audio sound files of the Disklavier parts allowed me to gradually learn how to pace these pauses and the changes in tempo, but given the number of these instances, and the variable pacing, complete and secure accuracy eluded me"*. The final stage was pre-concert rehearsals held the day before and on the day of the concert, which was the first opportunity he had to perform with the Disklavier. When asked prior to the performance whether he had any particular strategies for the performance Performer Z stated: *"At the moment I don't feel I have [control] except for the very beginning where it says clearly there's, you know options one, two, three. Other than that I don't feel I have any say in how it happens and have not really worked out exactly why it moves across [paths] or doesn't"*.

In contrast, **Performer A** was provided with a paper and digital versions of the score that showed all of the codes and also the software installed on her laptop. She also had access to a Disklavier and so throughout her preparation she moved from a MIDI keyboard to practice

alongside the Disklavier part, to her Grand piano, and then to sessions with the system and Disklavier. Performer A offered an in depth and enlightening reflection on her strategy: *"When first approaching Climb!, I was determined to interpret the notated score as accurately as possible. I saw the Climb! system as an absolute and consistently correct duo partner to whom I should make continuous adjustment."* However, through experience with the system she found that vagaries of tempo made perfect coordination impossible. She described how *"the frustration which emerged from the uncertainty of strictly synchronised entrances"* led her to experiment with musical variations in style, concluding that these represented a *"change in my relation to the Disklavier part from submissive to assertive"*. Given this experience, she observed that: *"After all, the full score of Climb! could be performed solely by the Disklavier and the result could be similar but with greater note accuracy. It can feel intimidating to perform on an instrument whose mechanics enable it to reach far greater technical aptitude than oneself. For me this was an invitation to exploit the "human" parts of my playing such as tempo rubato and dynamic fluidity"* (i.e., an expressive quickening or slackening of the tempo).

Unlike Performer Z, Performer A felt quite confident about the Challenge codes, their location in the sections and her ability to perform them. Before the performance she stated that she intended to deliberately perform some Challenge codes incorrectly in order to control her route through the work: *"Going to [play] Stones [i.e., path 2] and then down to [path] three and then across to [path] one to get a bit of everything ... well it's kind of upsetting if you've spent quite some time learning this ... and then you only get to play a third of it. So, I want to explore as much as possible."*

7.3 Playing the Score

To have a strategy is one thing, but to deliver it in a concert performance is quite another. Our performers adopted various tactics to deliver performances as they unfolded. We now present a series of vignettes that illustrate these, drawing upon interviews, audio and video recordings and system logs, to paint a rich picture of various incidents that occurred along the way, each of which involved a different notion of failure.

7.3.1 Vignette A: Failing a challenge leads to a surprise path

As previously detailed, the location of the Challenge codes within the score was deliberately withheld from Performer Z, and consequently any failures would be unintended. Z failed two codes in his performance, and we focus here on the first. In the section *Stones* the code requires the

performer to improvise. The score calls for a short melodic passage where the outer notes of the passage are written (i.e. the first three notes and the last one), but the inner notes are left open to the performer's choice and discretion. The 'code' definition uses Regular Expressions (a format typically used in computer science) to support flexible matching of codes [18]. This narrow, discreet and flexible detection mechanism deliberately avoids the sort of human error recovery found in some score following systems [11], which would constrain performers' scope to improvise around the score. Comparing the composer's score alongside Z's performance we observe that they started to improvise after the first note of the passage, rather than after the first group of three prescribed notes, thus breaking the code in the process. Z elaborates: *"This I am just playing wrong, although the phrase starts and ends on the correct notes. I think I probably got too carried away with the freedoms granted by the notation in second part of the phrase and created a whole phrase that, though largely improvised, still fits into the same harmonic colour."* The consequence of this failure was the need to quickly respond to a surprising jump to a new part of the score.

7.3.2 Vignette B: Playing on regardless

The Challenge codes are designed to (sometimes) be failed. But failing to play a Disklavier code means that the Disklavier will not play in that section, an omission that significantly disrupts the intended musical interaction between human and system. This in turn raises the issue of what a performer does if these codes pass unmatched. We witnessed two examples of Disklavier code failures that both took place in P1 (the dress rehearsal performance at the premiere of *Climb!*). Specifically, they were in sections *Path 1a* and *Path 1b* (see Figure 1). In *Path 1a* Performer M played the passage containing the Disklavier code an octave lower than written in the score, and as code matching is pitch register specific, the system did not recognise it. M continued to play without the accompanying Disklavier part for the remainder of the section. The possibility of failure/error is always present in a classical music performance and pianists are trained to deal with it. M recalled how a piano teacher from her early years had told her, *"Even if you'd be dying, just carry on playing!"*. No matter what happens, the performance has to continue, preferably so that the audience (or as many listeners as possible) will not notice anything. Fortunately, the human performer's part in this section consists of constantly flowing melodic patterns with little break in their rhythmic momentum, thus masking, at least to the ill-informed ear, the missing partner.

7.3.3 Vignette C: Improvising repetition to make further attempts

The examples above are the only occasions when a Disklavier code was not matched, and the performer carried on without the Disklavier accompaniment. But there were other near misses when a Disklavier code was initially failed but subsequently recovered. For example, in the section *Stones* in P1, Performer M played the Disklavier code twice, as their first performance of the code contained some note errors, successfully triggering the Disklavier on the second attempt. Similarly, in the section *Echo* from P9 Performer M performed the Disklavier code three times in succession before it matched, again due to performance error. In both instances the performer judged in the moment that a deviation from the score – looping back to play the same phrase again – would be less disruptive than losing the Disklavier part.

7.3.4 Vignette D: Musical interpretation avoids clashes

One of the qualities of the *Climb!* is that both the human and system performers occupy the same instrument. This opens up the potential for clashes, where both 'reach' for the same key at the same time. For A, this issue presented a number of times, most prominently in *Stones*. In this section the human and Disklavier cycle through similar descending interwoven phrases that occupy the same register of the keyboard. If these are performed out of sync they can result into an unmusical deformation of the phrases. To reduce the potential for the two performers tripping over each other, A chose to perform her phrases with staccato (i.e. each note sharply detached or separated from the other), thereby leaving more 'space' for the Disklavier to perform its notes. Although this performance instruction was not written into the score, she felt that the staccato articulation *"suggested the imagery of small stones falling down a path"* and so was suited to the section. So a subtle variation of the part as scored was justified musically and by the judgement that the duet with the Disklavier would be more reliable as a result.

7.3.5 Vignette E: Choosing to fail by improvising off the score

Analysis of the system logs shows, on the face of it, that A failed two Challenge codes in her performance, in sections *Stones* and *Herd of Cows*. She commenced her performance true to her pre-performance strategy, choosing Ending 2 in *Basecamp* and then failing the Challenge code in *Stones* section to branch off onto path 3. Post-performance, A discussed the uniqueness of the *Climb!* score and how she approached failing that Challenge code: *"It is not often that one is given an invitation to play wrong notes. However, when*

choosing to not execute the code correctly, I still wanted to keep the intention of the phrase intact and this was most easily done by altering only one note in the sequence whilst following the general intention.” The system logs support A’s description that she deliberately altered just the final note of the phrase to execute a branch onto path 3.

7.3.6 Vignette F: Failing to fail by not improvising enough

In contrast, there was at least one occasion when a performer attempted to fail a Challenge code that was nonetheless considered successful by the system. In the second performance, M planned to take a contrasting path to the first. Having previously played path 2, she intended to fail the Challenge code in *Tree Trunk* in order to jump to path 1 or 3, by altering the pitch formation of the code accordingly. However, the formation of the code in this section used regular expressions to enable some flexibility of matching and M failed to diverge sufficiently, and she was forced to continue playing path 2, resulting in the two performances being more similar than desired – she failed to fail.

7.3.7 Vignette G: Anticipating failure

The Climb! software system has generally performed without significant technical issues. However there have been occasional problems with the tempo of the Disklavier parts. In particular, during A’s performance in P5 (concert 2) the tempo and rhythmic playback of the Disklavier was inconsistent and at times significantly slower than normal. Analysing the system logs, the most notable example is found in the section Echo, where the Disklavier part took 50 seconds to play compared to 30 seconds for the other three performances where it was triggered. Performer A suspected that this inconsistent tempo was symptomatic of deeper problems: *“My feeling is when something starts playing up it’s very close to crashing altogether”*. Her pre-performance plan had been to branch over to path 2 at the penultimate section in order to perform *Birds Attack*, but because of the inconsistent tempo of the Disklavier she altered her plan and instead performed the Challenge code correctly to continue along the current path: *“I mean I really like Birds Attack. It’s actually probably my favourite section. And I wanted to do it but ... I skipped it because I didn’t want to ruin the piece”*.

7.3.8 Vignette H: Giving up and apologising

The last two vignettes focus on another concert performance–P4–which experienced the most significant system difficulties. We experienced some issues during the set-up and rehearsal prior to the performance, most notably

significant system ‘lag’ previously unexperienced. This lag resulted in all system actions experiencing high degrees of latency. The first attempt to perform was abandoned when the system became totally unresponsive towards the end of *Basecamp*, resulting in an apology to the audience.

7.3.9 Vignette I: Intervening from behind the scenes

Continuing on from vignette H, the performance was attempted again after a short delay during which the programme of works was re-ordered to enable another chance. However, given the significant delay, there was now insufficient time to deliver a full performance of *Climb!* by even the shortest possible path up the mountain. The solution was for a technician to monitor progress and manually trigger jumps through the score (possible via the system’s console) from behind the scenes so as to push the performer along an expedited path. A performance was delivered and *Summit* was reached in time, but afterwards Performer M felt that it did not count as a valid performance of the work and did not want it included in the official performance archive.

8 DISCUSSION: AESTHETIC FAILURE EXPLAINED

Climb! provides an opportunity to understand how highly skilled humans ‘fail’ aesthetically when performing with a very capable system. We now analyse our findings to reveal the complex and layered nature of failure; consider how failure may contribute to the aesthetic of performance; and relate the strategies and tactics we encountered above to each other. We recognise from the outset that failure is a subjective phenomenon: the performer, audience, composer and system might all have divergent views as to whether different aspects of a musical performance succeeded or failed. A performer might think that they have failed but the audience not notice, or an expert may spot flaws in a beginner’s playing that they would not notice themselves. In what follows we consider the performer’s view as this is what our data directly speaks to. Thus, we are interested in whether and how performers view particular interactions as failures. We begin by laying down some ‘definitional groundwork’ to help us reason about failure and its aesthetic consequences in a suitably nuanced way.

8.1 Layered Failure

We first consider what it means to fail in an interactive performance. We recognise four broad types of failure from our experience of Climb!:

- Failure that remains within the narrative of the work, e.g., when the performer fails the challenges that they encounter on the mountain.
- Musical failure that compromises the musicality of the performance in some way, for example in the performer’s musical interpretation and expression.
- Failure to perform a recognisable and acceptable version of this particular work (as scored); they might play good music, but it is not deemed to be ‘this work’.
- Failure to deliver anything that is recognisable as a performance at all, e.g., having to apologise to the audience and refund tickets (unless of course, this is part of the show!).

These are related in complex ways. An obvious layering is that a performer first needs to establish a performance in order to play the score, which in turn is necessary to deliver narrative success. However, our examples reveal other interesting cases. Vignette {A} for example, involved narrative failure, but successes in playing the scored work, musicality, and delivering a performance. Vignette {C} involved eventual success in playing the score but at the cost of a degree of musical failure. {H} failed to deliver a recognizable performance. {I} did eventually achieve this, but it was not deemed to be a valid rendition of the score (and so was not included in the official archive), though there were musical successes along the way. Thinking about failure as being layered in this way helps us understand that a single interaction may simultaneously involve both success and failure (at different layers) and enable us to unpick apparent oxymorons such as performers intentionally succeeding at failing {E} or even failing to fail {F} – both can happen when different layers of failure are involved.

8.2 Aesthetic Failure

Next, we consider the aesthetic consequences of failure. While failures can of course be catastrophic to varying degrees, from noticeable musical mistakes to irrevocable breakdown of the performance {I}, we have seen several ways in which they can enhance the aesthetic of a performance. Indeed, narrative failure is fundamental to the concept and aesthetic of *Climb!* and the work would be far less interesting if performers always succeeded at the challenges they encounter. The idea that failures are an important, even necessary, aspect of performance aesthetics is of course, not news. In *Aesthetics of Failure*, Cascone [8] brings our attention to the “detritus” or “by-product”–the failures of digital tools–in creating ‘glitch’ music, where

these digital artefacts become the genre, rather than just a facet of its creation, forcing us “to *examine our preconceptions of failure and detritus more carefully*”. While we agree that system failure can indeed contribute to the aesthetics of musical performance, our findings reveal further ways in which *human failure*–especially when interwoven with system failure–can also have aesthetic consequences including:

- Variation: in which narrative failure allows performers to vary their paths through the score to generate interest or fit particular timing constraints {E,I}.
- Musical interpretation: in which performers seek out alternative, but still suitably aesthetic, musical expressions (e.g., accommodate the timing of the Disklavier {D}).
- Improvisation: where performers have to improvise in response to the surprising results of narrative failure {A} or success {F} or have to play ‘off the score’ in order to deliberately fail a code {E} or improvise repetitions in order to have further attempts at a code {C}.
- Risk and liveness: while witnessing success or failure may not be the primary motivation for attending a musical performance, as it is with sport (although there are of course musical competitions), the risk involved in a skilled performer undertaking a difficult challenge adds to the frisson of live performance [33].

Having laid the necessary groundwork for reasoning about aesthetic failure in interactive performance, we are now in a better position to be able to compare human performers’ strategies and tactics for engaging with it. We now revisit our vignettes once more, but this time asking the questions: how did our performers approach failure? And how did they approach the system?

8.3 Approaches to Failure

Our study reveals that performers adopted three broad approaches towards failure. The first was to positively embrace failure as an opportunity to enrich the performance. A tactic included deliberately failing codes at the narrative level so as to introduce variation in terms of choosing a distinctive path through the piece or prolonging the performance to cover more of the score, while simultaneously improvising musical failure by coming up with a variant of a code that would sound right but fail to be matched {E}. This was a deliberate response to the composer’s intent to create a game-like performance. A second quite distinct way of embracing failure was seen in {A}, in which the pianist played the score without being

aware of the location of codes, allowing them an unconstrained opportunity for musical interpretation of the score, but then also having to respond to the surprise of jumping to different sections of the score at the narrative level depending on whether codes succeeded or failed.

A quite different and perhaps more obvious approach is to shun failure, treating it as a problem to be avoided and performing in a way that minimizes the possibilities. However, in *Climb!*, avoiding failure at one level can involve embracing it at another. We saw how Performer A changed her pre-performance plan at one point, deliberately failing to fail at the narrative level in order to avoid potential problems at the performance level due to her mistrust of the Disklavier's erratic behaviour {G}. An alternative tactic is shown in {D} in which the performer compromised their playing style, a degree of musical failure, in order to ensure that they could play the score along with the Disklavier.

A third approach is to mitigate failure, typically by trying to disguise it. As with embracing and shunning, this strategy operates across levels of failure. Vignette {C} reveals M attempting to recover from failure to perform the score by improvising repeat attempts in a way that they would not overtly appear to be musical failures. Sloboda [cited in 35], in discussing deviations from the score, noted that *“one learns to create an impression of accuracy in a performance”*. The more catastrophic failure of {}, potentially across all levels, required live orchestration from a human technician in as unobtrusive a manner as possible to salvage any success.

8.4 Approaches to Capable Interfaces

Performer A's striking account of asserting herself as a performer rather than submitting to the system reveals how human performers wrestle with how they should approach interaction with highly capable interfaces, such as the Disklavier; a challenge noted by McNutt [32] who asserts that the human performer is all too often corralled into a submissive position when performing with a computer partner. Her approach became one of assertion, finding the space or gaps in the system and making an expressive musical interpretation in response. The key weakness in the system was to be found in its timing, which was neither as reliable nor expressive as that of the human. This reflects previous ideas in HCI such as seamful design [7,9] or creatively exploiting the difference between expected and sensed actions [42]. Another form of assertion was to try to master the system, thereby being able to perform failures at the narrative level and so control it {E,G}. However,

submission might also be an aesthetically valid response to the system as shown by Z's approach of giving up any attempt to control it and instead improvising in response to its choices, however surprising {A}. Other examples show an approach that involved reaching more of a compromise with the system, in trying to assert control, but then sometimes losing control when things failed {C} or in compromising playing style to suit its capabilities {D}.

8.5 A Taxonomy of Strategies for Aesthetic Failure

So far, we have revealed how performers adopted different approaches to failure and to the system, leading to distinctive tactics for aesthetically embedding failure into their performances. We draw these together into an overarching taxonomy of aesthetic failure. Our aim is not to definitively capture failures as specimens under the microscope, or to reduce failure to a narrow set of properties; we believe that it is too rich, subjective and 'live' a phenomenon for this, and that there are certainly other interesting aspects of failure that we have not considered here (e.g., accountability to a wider audience to name just one). Rather our intention is to broaden HCI's conception of failure and so open up the design space to new creative possibilities. Our taxonomy is therefore a form of intermediate design knowledge that tries to bridge between a specific design instance (i.e., *Climb!*) and more general theory [23].

By comparing performers' approaches to failure with their approaches to the system we are able to separate out our vignettes and the tactics they employed, revealing how our performers did indeed explore a wide space of aesthetic failure. Reflection on this wider design space uncovers higher-level strategies that might guide performers and designers to aesthetically engage failure:

Game the system – in which performers embrace failure and assert themselves with the system, learning and mastering it so that they can creatively employ failure at the narrative and other levels to take creative control and fail in interesting ways, for example introducing variations.

Tame the system – in which performers assert themselves but with the intention of shunning failure, bringing the system under an acceptable level of control so that they can deliver a good performance.

Ride the system – in which they give themselves up to both failure and the system, accepting its consequences and improvising in response. Under this approach the system may take the performer to unusual places that demand creative responses.

Serve the system – in which humans become subsumed as a component of the system in order to help it succeed at its objectives, including cases in which human operators add intelligence to the system.

These however, are extreme points at the four corners of our taxonomy (see Figure 2). In practice performers will often seek out a balance between them, for example having to tame the system to a degree before they can then game it, or even moving between them as circumstances dictate. Thus, many instances of performance may lay closer to the centre in which performers *negotiate with the system* in a shifting and responsive way. Our corner strategies are markedly different from previous considerations of human-agent interaction as involving either co-allocation, co-operation and collaboration [6]. While such strategies may certainly be involved in negotiation at the centre of our taxonomy, they do not capture the creative tension and hence aesthetic potential that arises when humans and systems adopt more provocative stances towards each other.

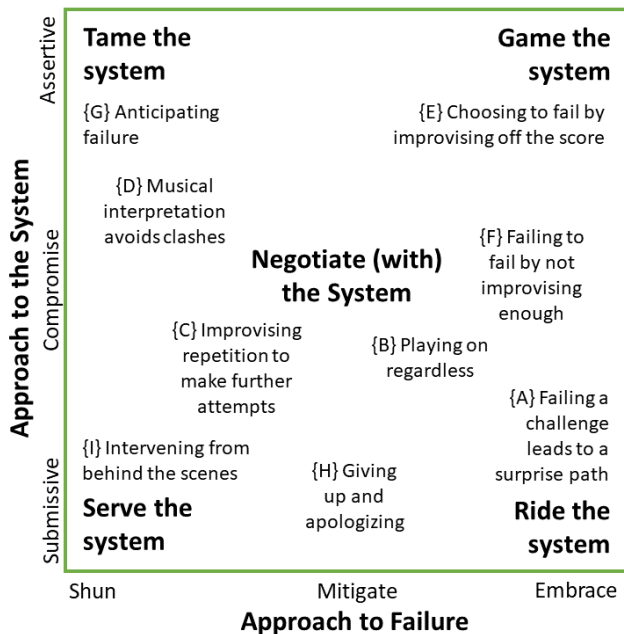


Figure 2: A Taxonomy of Aesthetic Failure

9 RETHINKING FAILURE IN HCI

What are the benefits of our taxonomy? On the one hand it provides sensitizing concepts [5] to guide the analysis of studies of performative interactions and potentially of failure in HCI more generally. On the other, it speaks to the design of future interactive systems. We consider three application areas in which these benefits might be realised.

9.1 Culture and entertainment

Perhaps most immediate benefit lies in cultural and entertainment computing. As noted earlier, there is a rich history of creatively exploring failure in the arts, including in experimental music. Our framework suggests ways in which failure might be considered to be an inherent part of more everyday musical interactions with digital musical instruments, performance management systems and musical tuition systems. Taking the latter as an example, embracing risk, failure and improvisation is, in our opinion, an underrepresented aspect of conventional music learning. For instance, contemporary musical tuition systems such as Yousician [43] reward accuracy of reproduction, but our framework raises the counterintuitive prospect that music tuition systems (that use embedded musical codes or similar technologies) might reward aesthetic failure as well as success, encouraging players to improvise variations that are ‘off the score’ in addition to moving them on to a new piece when they can successfully play key passages. There may also be benefits to games and sports that already depend on the possibility of failure in the game world, but might be extended to embrace failure—and our strategies towards it—at this and other levels. As games take on aspects of public performance, being played socially or as eSports [14], it becomes important to distinguish aesthetic success and failure of the performance from simple success and failure within the rules of the game. Similarly, creative uses of games platforms such as Machinima [30] may willfully disregard the norms of success within the game in favour of aesthetic renditions that convey something completely different from the original game.

9.2 Human-robot interaction

Robots are spreading into everyday life, from today’s semi-autonomous vehicles to future care, rescue and even musical robots. While the Disklavier in Climb! is not strictly an autonomous robot (it responds to scripted musical triggers and has limited capacity for movement), we argue that—as a physically actuated interface that feels to a performer like a partner in a duet—it speaks to the design of Human-Robot Interaction. HRI research has considered the importance of designing for flexible autonomy and mixed initiative interactions [17] and allowing for varying degrees of autonomy as humans tighten and loosen the reins of control [16,18] as well as ethical discussions of robots deskilling, replacing [29] or infantilising [39] humans (concerns directly mirrored in the use of instruments such as Disklaviers to replace human musicians in bars and hotel lobbies). Our framework can widen HRI’s agenda to better accommodate or even embrace aspects of failure. We

illustrate this with an anecdote drawn from the experience of one of our authors. In a recent incident, their car, which is capable of autonomous parallel parking, successfully parked itself in a tight space: a technical success. However, it took many cycles of inching back and forward to achieve this while a queue of waiting cars built up behind. So at another level this was an aesthetic and social failure, in which the human driver literally ‘rode’ the system. As well as highlighting (once again) that success and failure can occur simultaneously at multiple levels, it also shows that there may be an important aesthetic elements to even the most mundane of activities. We suggest that similar considerations will apply to more complex HRI, and that strategies such as taming, gaming, riding, serving, or negotiating may help humans engage in more creative interactions with robots. In turn, robots’ own internal models of human action and intent might be enriched to reason about how humans approach failure, for example that they sometimes intend to simultaneously succeed and fail (at different layers) for good reasons. We also raise the challenging question of whether autonomous systems should ever provoke failure in humans? While this may sound like an uncomfortable proposition, we raise the question of whether truly creative relationships with future robots may require us to re-consider our framework from the *robots* perspective: should the robot ever game, ride or tame the human?

9.3 Conversational interfaces

Conversational interfaces that employ natural language processing are a further form of autonomous system that warrants consideration. Humorous uses of language that employ irony, sarcasm and teasing often involve sophisticated wordplay that deliberately invokes apparent failures of language. For example, double meanings in puns rely on ambiguous statements that simultaneously fail and succeed semantically. Indeed, linguistics has accounted for irony, sarcasm and teasing in terms of the participants in a conversation simultaneously inhabiting multiple layers so that they can make apparently false or nonsensical statements at one (narrative) layer while simultaneously understanding that they are doing this in a second (social) layer [22]. We argue that humorous wordplay is an important feature of human language that relies on aesthetic and layered failure. Failed interactions with Alexa, as just one example, can become a ‘laughing matter’ for participants who tease the system for its failings [34], reflecting how humour can smooth over awkward social moments while enabling social bonding. Strategies such as ‘gaming’ arise when participants acquire a sufficient

mastery of language along with an assertive attitude to the system and positively embrace failure as an opportunity for wordplay. In short, we propose that our taxonomy can inform the challenge of introducing humour into conversational interfaces so that they too can engage humans in a more creative and aesthetically appropriate manner.

10 CONCLUSION

We have revealed how failure in live performance is both a complex and aesthetically important matter. Failure can be considered as being multi-layered, with failure at one level often relying on success at another and vice versa, leading to variation, improvisation and liveness. We have revealed how humans can adopt diverse strategies to responding to and also deliberately invoking failure in interaction, potentially gaming, taming, riding and even serving the system and have proposed that such strategies may benefit several areas of HCI including cultural and entertainment computing, human-robot interactions and conversational interfaces. We encourage researchers and practitioners to consider failure as both a layered and aesthetic matter and to be open to creative strategies for engaging with failure that: enable humans to better assert their creativity with increasingly capable systems; enable autonomous and intelligent systems to better reason about the complexities of failure; and ultimately design systems that are themselves capable of failing in appropriate and aesthetic ways.

ACKNOWLEDGMENTS

This work was supported by the UK Engineering and Physical Sciences Research Council [grant numbers EP/L019981/1, EP/M000877/1] and the Kone Foundation. The data underlying this paper is included within the paper.

REFERENCES

- [1] Ross J. Anderson. 2010. *Security engineering: a guide to building dependable distributed systems*. John Wiley & Sons.
- [2] Steve Benford, Matt Adams, Nick Tandavanitj, Ju Row Farr, Chris Greenhalgh, Andy Crabtree, Martin Flintham, Brendan Walker, Joe Marshall, Boriana Koleva, Stefan Rennick Egglestone, and Gabriella Giannachi. 2013. Performance-Led Research in the Wild. *ACM Transactions on Computer-Human Interaction* 20, 3: 1–22. <https://doi.org/10.1145/2491500.2491502>
- [3] Steve Benford, Chris Greenhalgh, Adrian Hazzard, Alan Chamberlain, Maria Kallionpää, David M. Weigl, Kevin R. Page, and Mengdie Lin. 2018. Designing the audience journey through repeated experiences. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, 568. <https://doi.org/10.1145/3173574.3174142>
- [4] S M Astrid Bin. The Show Must Go Wrong: Towards an understanding of audience perception of error in digital musical instrument performance. (Doctoral Dissertation, Queen Mary University of London). 368.

- [5] Glenn A. Bowen. 2006. Grounded theory and sensitizing concepts. *International journal of qualitative methods* 5, 3: 12–23. <https://doi.org/10/gcdvbt>
- [6] Jeffrey M. Bradshaw, Paul J. Feltoovich, and Matthew Johnson. 2012. 13 Human-Agent Interaction. *The Handbook of Human-Machine Interaction: A Human-Centered Design Approach*.
- [7] Gregor Broll and Steve Benford. 2005. Seamless design for location-based mobile games. In *International Conference on Entertainment Computing*, 155–166. https://doi.org/10.1007/11558651_16
- [8] Kim Cascone. 2000. The aesthetics of failure: “Post-digital” tendencies in contemporary computer music. *Computer Music Journal* 24, 4: 12–18. <https://doi.org/10/bmb6ts>
- [9] Matthew Chalmers, Ian MacColl, and Marek Bell. 2003. Seamless design: Showing the seams in wearable computing. <https://doi.org/10/fjrdw4>
- [10] Cello Fortress - A game by Joost van Dongen. Retrieved January 3, 2019 from <http://www.cellofortress.com/>
- [11] Arshia Cont, José Echeveste, Jean-Louis Giavitto, and Florent Jacquemard. 2012. Correct automatic accompaniment despite machine listening or human errors in Antescofo. In *ICMC 2012-International Computer Music Conference*.
- [12] Lowell Cross. 1999. Reunion: John Cage, Marcel Duchamp, Electronic Music and Chess. *Leonardo Music Journal* 9: 35–42. <https://doi.org/10/cf3f2j>
- [13] Anna Dickinson, John Arnott, and Suzanne Prior. 2007. Methods for human-computer interaction research with older people. *Behaviour & Information Technology* 26, 4: 343–352. <https://doi.org/10/fwzwtwh>
- [14] Guo Freeman and Donghee Yvette Wohn. 2017. eSports as an emerging research context at CHI: Diverse perspectives on definitions. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, 1601–1608. <https://doi.org/10.1145/3027063.3053267>
- [15] Anders Friberg. 2006. pDM: an expressive sequencer with real-time control of the KTH music-performance rules. *Computer Music Journal* 30, 1: 37–48. <https://doi.org/10/d4krt9>
- [16] Kenneth H. Goodrich, Paul C. Schutte, Frank O. Flemisch, and Ralph A. Williams. 2006. Application of the H-mode, a design and interaction concept for highly automated vehicles, to aircraft. In *25th Digital Avionics Systems Conference*, 2006 IEEE/AIAA, 1–13. <https://doi.org/10.1109/dasc.2006.313781>
- [17] Michael A. Goodrich and Alan C. Schultz. 2008. Human-robot interaction: a survey. *Foundations and Trends® in Human-Computer Interaction* 1, 3: 203–275. <https://doi.org/10/b9k4td>
- [18] Chris Greenhalgh, Steve Benford, Adrian Hazzard, and Alan Chamberlain. 2017. Playing fast and loose with music recognition. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, 4302–4313. <https://doi.org/10.1145/3025453.3025900>
- [19] Official Site of Guitar Hero. Retrieved January 3, 2019 from <https://www.guitarhero.com/uk/en/>
- [20] Guy Hoffman and Gil Weinberg. 2011. Interactive improvisation with a robotic marimba player. *Autonomous Robots* 31, 2–3: 133–153. <https://doi.org/10/d993sp>
- [21] Stephen A. Hedges. 1978. Dice music in the eighteenth century. *Music & Letters* 59, 2: 180–187. <https://doi.org/10/bgjcpg>
- [22] Herbert H. Clark. 1996. Using language. 1996. Cambridge University Press: Cambridge 952: 274–296. <https://doi.org/10.1017/s0022226798217361>
- [23] Kristina Höök and Jonas Löwgren. 2012. Strong concepts: Intermediate-level knowledge in interaction design research. *ACM Transactions on Computer-Human Interaction (TOCHI)* 19, 3: 23. <https://doi.org/10/f225d4>
- [24] Ioanna Iacovides, Anna L. Cox, Patrick McAndrew, James Aczel, and Eileen Scanlon. 2015. Game-Play Breakdowns and Breakthroughs: Exploring the Relationship Between Action, Understanding, and Involvement. *Human-Computer Interaction* 30, 3–4: 202–231. <https://doi.org/10/gfsmxz>
- [25] Jesper Juul. 2013. *The art of failure: An essay on the pain of playing video games*. MIT Press. https://doi.org/10.1162/leon_r_00879
- [26] Maria Kallionpää. 2016. Performing the super instrument: reaching beyond technical and expressive capabilities. In *Proceedings of the conference on Electronic Visualisation and the Arts*, 28–33. <https://doi.org/10.14236/ewic/eva2016.7>
- [27] Maria Kallionpää and Hans-Peter Gasselseder. 2016. The imaginary friend: crossing over computer game scoring techniques and musical expression. In *Proceedings of the conference on Electronic Visualisation and the Arts*, 42–48. <https://doi.org/10.14236/ewic/eva2016.9>
- [28] Nina Keith and Michael Frese. 2008. Effectiveness of error management training: a meta-analysis. *Journal of Applied Psychology* 93, 1: 59. <https://doi.org/10/fbksqf>
- [29] Patrick Lin, Keith Abney, and George Bekey. 2011. Robot ethics: Mapping the issues for a mechanized world. *Artificial Intelligence* 175, 5–6: 942. <https://doi.org/10/fxhj4b>
- [30] Henry Lowood. 2008. High-performance play: The making of machinima. *International Journal of Technology Management & Sustainable Development* 7, 1: 25–42. <https://doi.org/10.1386/jmpr.7.1.25/1>
- [31] Clemens Maidhof, Martina Rieger, Wolfgang Prinz, and Stefan Koelsch. 2009. Nobody is perfect: ERP effects prior to performance errors in musicians indicate fast monitoring processes. *PLoS One* 4, 4: e5032. <https://doi.org/10/bj5stj>
- [32] Elizabeth McNutt. 2003. Performing electroacoustic music: a wider view of interactivity. *Organised Sound* 8, 3: 297–304. <https://doi.org/10/ddk46k>
- [33] Imogene Newland. 2013. Embodying failure: Music performance, risk and authenticity. *Somatechnics* 3, 1: 98–118. <https://doi.org/10/gd7d5v>
- [34] Martin Porcheron, Joel E. Fischer, Stuart Reeves, and Sarah Sharples. 2018. Voice Interfaces in Everyday Life. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, 640. <https://doi.org/10.1145/3173574.3174214>
- [35] Bruno H. Repp. 1996. The art of inaccuracy: Why pianists’ errors are difficult to hear. *Music Perception: An Interdisciplinary Journal* 14, 2: 161–183. <https://doi.org/10/gd7d5x>
- [36] Jean-Claude Risset and Scott Van Duyne. 1996. Real-time performance interaction with a computer-controlled acoustic piano. *Computer music journal* 20, 1: 62–75. <https://doi.org/10/d9bv5b>
- [37] William Ryan and Martin A Siegel. Evaluating Interactive Entertainment using Breakdown: Understanding Embodied Learning in Video Games. In *DiGRA Conference*. 2009. 9.
- [38] Martina Angela Sasse, Sacha Brostoff, and Dirk Weirich. 2001. Transforming the ‘weakest link’—a human/computer interaction approach to usable and effective security. *BT technology journal* 19, 3: 122–131. <https://doi.org/10/cj5hm4>
- [39] Amanda Sharkey and Noel Sharkey. 2012. Granny and the robots: ethical issues in robot care for the elderly. *Ethics and information technology* 14, 1: 27–40. <https://doi.org/10/cdd62t>
- [40] SingStar. Retrieved January 3, 2019 from <https://www.singstar.com/home.html>
- [41] Plurality Spring by Paul Turowski and Simon Hutchinson. Retrieved January 3, 2019 from <https://batta.itch.io/plurality-spring>
- [42] Lucy A. Suchman. 1987. Plans and situated actions: The problem of human-machine communication. *Cambridge university press*. <https://doi.org/10.1017/cbo9780511808418.011>
- [43] Yousician – Your personal music teacher. Retrieved January 7, 2019 from <https://yousician.com/>
- [44] Dieter Zapf, Michael Frese, Felix C. Brodbeck, C. Graf Hoyos, D. Frey, and D. Stahlberg. 1999. Fehler und Fehlermanagement. *Arbeits-und Organisationspsychologie*.
- [45] John Zimmerman, Jodi Forlizzi, and Shelley Evenson. 2007. Research through design as a method for interaction design research in HCI. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, 493–502. <https://doi.org/10.1145/1240624.1240704>