Musical Grid Interfaces: Past, Present, and Future Directions

Beat Rossmy¹, Alexander Wiethoff¹

¹LMU Munich

Published on: Apr 29, 2021
License: Creative Commons Attribution 4.0 International License (CC-BY 4.0)
ABSTRACT

This paper examines grid interfaces which are currently used in many musical devices and instruments. This type of interface concept has been rooted in the NIME community since the early 2000s. We provide an overview of research projects and commercial products and conducted an expert interview as well as an online survey. In summary this work shares: (1) an overview on grid controller research, (2) a set of three usability issues deduced by a multi method approach, and (3) an evaluation of user perceptions regarding persistent usability issues and common reasons for the use of grid interfaces.

Author Keywords

History, Grid, Interfaces

CCS Concepts

• Human-centered computing → Interaction design → Interaction design theory, concepts and paradigms;

Figure 1

Today many musical instruments and controllers use grid interfaces in their designs. As grids we define illuminated button matrices.
Introduction

When considering the landscape of musical interfaces, even New Musical Instruments (NMI) often rely on established canonical interface forms. These include among others the keyboard, which is the common interface concept of instruments such as the piano, the organ or the accordion, and is as well commonly used in synthesizers, workstations, and samplers.

While the NIME community constantly creates new prototypes such as body controlled \cite{1,2}, tangible \cite{3,4}, or augmented \cite{5,6} instruments, these concepts can be regarded as innovative interaction models, but fail to establish themselves as canonical interface archetypes in the long run. However, when looking into the current palette of commercial NMIs one can observe a new interface form rising over the last 15 years. More and more instruments contain as one of their main interface elements an illuminated button matrix, which we hereafter refer to as the grid.

The first publications concerning this interface type in the musical context were published by Nishibori and Iwai in 2005 and 2006 \cite{7,8}. Since then, over 225 musical devices have been released on the market that use this interface concept in their designs and the amount is constantly rising (cf. Figure 2).

To discuss this shift within the scientific community, we present: (1) an overview on the development of grid devices, (2) a set of usability issues deduced by a multi-method approach, and (3) an evaluation of the user perception regarding these usability issues and general reasoning for the use of grids.
Related Work

In the following we provide a definition of grid interfaces for the context of this paper and present an overview of grid research and developments.

Grid Definition

While previously [9] musical soft- and hardware systems have been named “grids”, we exclusively contemplate hardware interfaces, which we consider to be the musical equivalent of Low Resolution Lighting Displays (LRLD) [10]. While both are light-matrices that represent information with only a few pixels [11][12][13], in grids these are literally the touch-point for interaction, which depict and enable the manipulation of information. We use the following three properties for our definition of grids in the musical context:

1. Rectangular and irregular low resolution matrices with visible cell size of $m \times n$ ( $m \in \mathbb{N}_{>1}, n \in \mathbb{N}_{>1}$ )

Figure 2
Timeline of grid device releases (data set). The graph shows the annual number of devices using a grid as a UI element, separated in five product categories.
2. Cells **physically stand out** from the interface surface
3. Cells are used for **in- and output**, but interaction and feedback are decoupled

Thus, we exclude (1) single buttons and row/column vectors as commonly used in step sequencers, (2) non-illuminated buttons as found in many drum pads as well as (3) touch screens [14] due to their lack of haptic buttons.

**Keys, Buttons, and Button-Matrices**

The usage of keys and buttons has a long tradition in musical instruments starting in the late 3rd century BC \(^1\). Keys were used to trigger the sound generation or to operate instruments via levers and other mechanics. They were commonly laid out linearly, but later non-linear layouts were conceptualized [15] and diatonic and chromatic **button matrices** were used in different instruments. Expanding on ideas such as **microtonality** and alternative playing techniques, matrix-like layouts, called isomorphic keyboards, got more popular [16][17].

Since 1963 push-button matrices were used in electronic devices (cf. phones) and research [18] and development (US3676607A) created new technologies which influenced the design of later grid interfaces. Early on, keys were used multifunctionally within telephones to enter both numbers and letters. After production, however, attached labels and analog components are permanent and prevent subsequent adaptation of the functions.

Illumination has been later added to buttons to highlight them in the dark or to redundantly indicate their state [19][20]. Next\(^2\), they were used for decoupled feedback [21], which allowed information to be provided to system operators on demand in control units with illuminated control panels (cf. NASA).

Since the 1980s, non-illuminated button matrices were used as musical interfaces for finger drumming and sequence programming. The Roland SP-808 can be considered as one of the first examples of such a multifunctional LED-button matrix used in a commercial instrument (see Figure 2).

**NMI Design: Instrument vs. Controller**

Concerning interfaces in the context of musical instruments, we refer to the first layer of the New Musical Instrument (NMI) \([22][23]\) framework, which distinguishes between (1) the **control layer** which is connected via the (2) respective **mapping** to a (3) **sound source**.
Acoustic instruments provide an implicit mapping, e.g. such as the mechanics between the keys (interface) and strings of a piano (sound source) directly transmit and translate key interaction to sound generation; electronic instruments, on the other hand, have to explicitly design this mapping [24]. For this work, we use the term instrument for devices which provide a mapping as well as a sound generator and the term controller if the device only offers the interface layer.

This is important since grid NMIs are "compositional as much as they are about physical performance" (Peter Kirn). Due to these different usage scenarios, manufacturers and users conceptualize and approach grid design differently. From a blank canvas (controller) which is "much more of a tool than an instrument" (Ed) all the way to in a NMI embedded and integrated interface.
Developments of Grid Interfaces in the Musical Context

We provide a chronological overview of the interfaces described in the following section (see Figure 3).

The first interfaces which extensively explored the capabilities of musical grids were the Tenori-On [7][8] and the monome grid [25]. While the Tenori-On is a self-contained instrument which uses a button matrix for tactile input and visual output, the monome grid is a generic controller for sound-applications. Thus, these devices represent the two fundamental NMI design strategies: (1) the grid as an instrument, a symbiosis of
interface and sound source, and (2) the grid as a controller, a from the sound source decoupled, freely assignable control surface.

In both cases the grid is the central interface which can be varied in different properties. The size of the button matrix (grid estate) determines the amount of information to be accessed and displayed. The button shape defines the tessellation pattern, which affects the layout and organization of the information being displayed. Mainly platonic tessellation patterns are used, such as triangular [26], rectangular [25], and hexagonal [27] patterns. The button shape can be domed [7], flat [28], or nearly integrated [29] in the surface, being only texturally distinguishable.

While initially, monochrome and binary-state LEDs were used, later, various-brightness [7], bi-chromatic [30] and RGB-LEDs [31] improved the color feedback. Though the extended color space provides more details, complex information is still hard to convey [10]. Thus, additional output elements such as screens [30] were used to enable actions on the device, previously only applicable on a computer. Further, shape-display technology [32] used the cell height as an output modality [33][34] expanding the former flat grid into the third dimension.

Regarding input interactions, grids adopted features such as: continuous perpendicular input (velocity and aftertouch [29]), planar interactions (vibrato, bending, sliding [28]), constructing shapes from modular grid elements [26], and the manipulation of the grid’s 3D-shape [33][34]. Other devices further added additional control elements such as knobs, sliders, antennas [35], and touch screens [30] to grid devices to allow interactions beside the perpendicular actuation of buttons. Most recent explorations add capacitive touch technology to the silicone button surface which enables interactions such as swiping [36].

While most grids are stand-alone devices, grids were also embedded into instrument-like [37] designs which affects the body position and the movement patterns during play, into modular designs to adapt to user needs [35], and have been further used in collaborative settings [38] which take into account group dynamics and collaborative musical processes.

**Taxonomy of Grid Interfaces**

We collected and reviewed a comprehensive data set of commercial grids, using a pearl growing search method [39] applied to images, web pages, and forum comments[^3] (start terms: “grid controller”, “button matrix”). In summary 228 grid devices (see
Figure 2) were collected, released between 1998 and the beginning of 2021. The collected device specifications are available as a data set which can be used as a source by other researchers investigating this topic (data set).

We identified seven design dimensions and the associated characteristics (cf. D1-D7 in Figure 4) based on the related work and the above mentioned data set. These are in the following considered to identify grid usability issues.

<table>
<thead>
<tr>
<th>DIMENSION</th>
<th>CHARACTERISTICS</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1 Grid Estate</td>
<td>Grid Estate = Width * Height (Pixels)</td>
</tr>
<tr>
<td>D2 Tessellation Pattern</td>
<td>triangles [26], rectangles [25], hexagons [27]</td>
</tr>
<tr>
<td>D3 Button Shape</td>
<td>domed [7], flat [28], structural [29]</td>
</tr>
<tr>
<td>D4 add. Control Elements</td>
<td>buttons, faders, encoders, joysticks, touchscreens, shape-changing surface, IR-sensors</td>
</tr>
<tr>
<td>D5 add. Output Elements</td>
<td>LEDs, screens, shape-changing surface</td>
</tr>
<tr>
<td>D6 Color Feedback</td>
<td>monochromatic, bi-chromatic, RGB</td>
</tr>
<tr>
<td>D7 Spatial Integrity</td>
<td>static, adaptable, modular</td>
</tr>
</tbody>
</table>

Figure 4  
Design dimensions of musical grid interfaces.

**Hardware**

The button matrix of a grid can be classified by the grid-estate, the tessellation pattern and the button design. These features in combination shape a grid’s primary appearance. We found that the grid resolution ranges from 2*2 up to 32*16 and has on average 8*4 buttons (median). About 98% of all devices use a rectangular and only 2% a hexagonal tessellation pattern. 50% of the devices integrate a display into their design, 83% use additional buttons, and 79% use other input elements such as keys, encoders or faders. We further found that 25% of all collected devices integrated multiple grids in one device. Common examples would be DJ equipment, which, based on the idea of a two deck system, provides two sets of the same controls. But other concepts were identified, such as devices which provided a narrower matrix for clip control and finger drumming as well as a wider one for sequence programming. Only 11% of all devices can be classified as grid-only controllers, i.e. that they only offer a button-matrix as their interface and no additional controls beside that.
While the classification of a grid-only device is unambiguous, other device types are rather classified in a continuum depending on the amount of additional interface-elements (see Figure 5). Devices having a stronger focus on the grid as their main interface can be classified as extended grids, if the grid becomes less important it is rather considered a grid extending a NMI.

As stated by a user, this different form factor provided by the hardware properties of grid-only devices and the resulting interaction implications achieve "a break from the comfortable tyranny from the piano keyboard" (Stretta).

**Input Interactions**

While all hardware properties define a grid's overall affordance, the performed interactions primarily focus on the buttons, which “afford pressing, but the grid itself affords carrying, holding, and a lot more” (emenel). Standard interactions with the matrix include pressing and holding as well as press and button combinations (double click, shift, combination within matrix). These are performed single-/two-handed, using the thumbs and all fingers. Grids are mostly used stationary and occasionally handheld.

By reviewing several hours of grid performances via video platforms, we identified advanced interactions, such as (1) strumming the buttons, (2) playing multiple grids at once, and (3) gestural input via proprietary hardware (leap motion, wii controller).
Grid interactions can be categorized as **Live Playing** (action and sound are directly coupled), **Conducting** (actions and sound are indirectly coupled, cf. triggering loops or drum-rolls), and **Operating** (actions and sound are not coupled). These categories reflect in the product types we identified which include: Instruments, Keyboards, Controllers, DJ-Mixers, Sequencers.

**UI Design**

The visual design of a grid application naturally implies the available interactions to the user. Following Norman [40], "*the illumination patterns are signifiers of the grid and the application that is being used with the grid*" ([papernoise](http://papernoise.com)). Without visual feedback, a button matrix can only indicate interactions statically (labels, icons). The light feedback allows for adaptive signifiers based on modes, context, or time. Thus, the interface becomes understandable beyond its physical affordance. Themes in grid UIs are:

1. **Physicality**: Objects on the grid behave like objects in the physical world. Lights move due to forces, are reflected of obstacles or trigger events based on collisions. E.g. the Tenori-On offered different modes "*that modeled physical processes including a bouncing ball*" ([Chris Mayes-Wright](http://www.chrismayesthesounds.com)).

2. **Familiarity**: The grid imitates musical interfaces. Columns represent faders, or cells function as keys/frets. Musicians can apply their prior-knowledge (finger-placement, chord shapes) but are not restricted to physical limitations (multiple notes on one string, pitch intervals).

3. **Direct Transfer**: Digital UI elements (cf. Ableton Live's clip matrix) are displayed on the grid. Colors are used to support orientation between the screen and the grid. Also older metaphors such as tapes and play/record heads can be found in grid applications.

**Usability Issues and Solutions**

We identified usability issues by (1) analyzing the presented data set using a theory driven approach in respect to usability heuristics [41], (2) backed the analysis with statements found in user forums, and (3) discussed the issues during expert interviews (n=4). We found the following usability issues:

(U1) **Limited Information Transfer**

While the visual properties of grids improve the overall information transfer [25], users struggle with limitations such LRLDs pose. They can comprehend the information
when creating projects, but encounter problems when either approaching unfamiliar apps (old, third-party, etc.) (andrew) or learning about new apps “by only looking at the video” (axel). Since only up to 12 colors should be used to present distinguishable information [42], grid users additionally “label their devices” (gli), manufacturers add icons or use displays to provide complex information in addition to LED colors. While these are common LRLD issues, it is assumed that regular usage [11] mediates these problems and that users can “even get to the point of muscle memory” (Peter Kirn), however the often generic nature of grids stands in contradiction to this point, since the context and thus the related interaction patterns can change dramatically.

(U2) Limited Control

Making interfaces more expressive is a common HCI problem [43][44]. While many users are comfortable with binary input actions “using the aspect of time” (Peter Kirn), others miss features such as velocity and after-touch and accompanying interface-elements such as “controllers [and] more knobs” (Wannop). Thus, manufacturers add UI elements (D4) to create an optimized workflow for specific applications (cf. Ableton Push [45]). This limits the generic nature and adaptability, but otherwise grids would remain restricted regarding many interactions. While, some typical controls such as faders can be imitated by rows or columns of grid cells, others such as rotary encoders [46], can’t be replicated with buttons.

(U3) Limited Grid-estate

The challenge of app design for LRLDs and low-res devices lies in the effective use of the available space [10]. Furthermore, grids not only passively display but actively provide access to information. The space can either be (1) limited by a small resolution (karol), (2) obstructed by infrequently used UI elements, or (3) is unsuitable for a larger and more intuitive UI (scanner). Since static UI elements compromise on the grid-estate, two strategies can be observed: (1) external buttons are used for additional functions (e.g. shift, navigation) or (2) the top row is permanently allocated for these. Our analysis showed that while some grids provide high resolutions such as 32*16 for which the reduced grid estate might not be an issue, most grids only provide about 8*4 buttons, this highlights the need for an effective use of the available space.

Cross-domain Driven Solutions

To approach the presented usability issues, we considered other HCI domains to inform solutions for the grid context, which enhance the capabilities without compromising their generic character.
**P1:** While we excluded interfaces such as the Lightpad Block or the Erae Touch from the grid device data set, due to their continuous interface surface, we acknowledge a promising approach in increasing information density by offering **multiple LEDs** or even small **screens** per cell. Thus, iconographic labels could be displayed, as already explored in the context of computer keyboards [47], or even Gestalt Psychology [48] can be used to visually bridge the gaps in the physical interface to highlight connected UI-elements.

**P2:** Since grid cells are limited regarding many interactions, radar chips [49], which are capable of tracking **midair gestures**, can be used to expand user control. Finger gestures such as turning an invisible knob could be used to adjust values represented by grid cells. Further, input expressiveness can be increased by considering flexible surface UIs [50][51]. Thereby, **3D interactions** can enable the manipulation of grid data or allow for more expressiveness.

**P3:** Since the screen real-estate problem is well known for mobile devices, we propose that concepts developed for this domain should be explored in the grid context. Force dependent gestures [52][53], posture dependent gestures [54] or edge/bezel interaction [55][56][57] are used to access additional features on demand. **Touch gestures** provide a rich resource of ubiquitous and familiar interactions that can be transferred to grids.

**Why to Use Musical Grid Devices?**

To investigate reasons for the use of grids as musical interfaces, we conducted an online survey with grid users (n=26). The survey covered participants' (1) opinion on usability issues and (2) general reasons for using grid devices. The questions were synthesized from expert interviews (n=4) and considered six design opportunities of NMI's [58] (see Figure 7). All questions were rated via 7-point Likert scales. Further, participants provided qualitative text answers.

**Participants**

Participants were recruited via the monome forum (llllllll.co), due to its expertise-wise homogeneous user base. The 26 participants (22 male, 3 unspecified, 1 non-binary; 5: 20-29 years, 12: 30-39 years, 9: 40-60+ years) were experienced in music making (5-44 years) and grid usage (1-13 years). Two of the participants did not own a grid when they took the survey. The rest owned one or multiple devices. As expected, 24 participants owned at least one monome grid. The devices owned, covered **open source hardware, generic controllers, instruments, software controllers**, and
sequencers. All participants agreed that their data would be used in anonymized form for scientific publications.

Restrictions and Limitations of Grids

During the expert interview we collected seven usability issues of grid devices in comparison with other music technologies (see Figure 6) and phrased statements based on these (I1-I7). During the survey, participants had to indicate their agreement with the statements. Our study indicated that the issues created a broad response, ranging from complete agreement (7) to complete disagreement (1). This has been indicated by high Interquartile Range (IQR) values [59]. Overall, the participants tended to disagree with the statements I1-I6. Participants further added the following issues:

1. cheap hardware quality and poor design
2. required expertise
3. poor application design and documentation
4. poor software integration

One participant explicitly stated to disagree with I1-I2 since “these are positives, and one of the reasons to use the grid” (ID4). When comparing grids to touchscreens as an alternative technology, participants stated haptics (n=19) and other general properties (n=6) such as the portability, durability, tactual experience, and minimalism as advantages. Further, 5 participants clearly appreciated the creative constraints due to properties we framed as issues.
Reasons to Use Grid Devices

Based on the free text answers, we deduced the following reasons to use and positive aspects of grid devices:

**Openness and Adaptability.** (n=16) Grid devices are often reprogrammable, flexible in use, and stimulate experimentation. This was considered to be important, since “other MIDI input device formats like keyboards are hard to use in a context outside of traditional note input” (ID9), further they “provide an open platform” (ID10) for self-defined musical processes.

**Haptics and Immediacy.** (n=10) Grid systems create an enjoyable tactile experience since they are “appealing instruments, both nice to look at and pleasant to touch” (ID17) and further allow users to feel and “push down multiple buttons at once” (ID6) which gives immediate access to the digital information.

**Design and Constraints.** (n=7) Restrictions such as the limited resolution or feedback define the grid experience and are even “enjoyed embracing the constraints of the device in creative ways” (ID6). Users who design their own applications are guided by narrowing down the possibility space.

**Compatibility and Community.** (n=7) Since grids often use common and open protocols such as OSC [60] or MIDI they are compatible with various soft-/hardware.
Further, user's found the strong community involved to be crucial since many are not programmers and thus the support and creative stimulation are essential.

**Approach and Metaphors.** (n=6) Grids are not an end in themselves, but rather are defining for the “interaction with these processes [and the] performance practice” (ID10). They are seen as fitting for many interactions such as programming patterns in the time domain and are building on metaphors or thinking structures of the users.

These reasons as well reflect the most supported usage opportunities (see Figure 7). While O2-O6 were supported, especially O2, O3, and O5 created high agreement. Contextualizing these with the deduced reasons, O2 and O3 support Openness and Adaptability, whereas Haptics and Immediacy is supported by O5. It is interesting, that 19 participants stated to appreciate the haptic qualities of grids, but O1 which is concerned with the grids’ “object character” only generated a neutral response.

<table>
<thead>
<tr>
<th>OPPORTUNITY</th>
<th>LIKERT RATING</th>
<th>MED.</th>
<th>IQR</th>
</tr>
</thead>
<tbody>
<tr>
<td>O1  I appreciate the “object character” (haptic qualities) of a grid controller.</td>
<td>![Likert Scale]</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>O2  I appreciate the “object character” (haptic qualities) of a grid controller.</td>
<td>![Likert Scale]</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>O3  I appreciate grids because the function as a “creative extension” to my workflow.</td>
<td>![Likert Scale]</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>O4  I appreciate grids since they create an element of “experimentation and randomness” in my music.</td>
<td>![Likert Scale]</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>O5  I appreciate grids since they offer “directness” regarding the control of otherwise intangible properties of a music application.</td>
<td>![Likert Scale]</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>O6  I appreciate grids since they provide “individualism and uniqueness” to my music setup.</td>
<td>![Likert Scale]</td>
<td>5</td>
<td>3</td>
</tr>
</tbody>
</table>

![Figure 7]

**Findings**

Looking back at the collected data set and presented resources, we derive two main aspects, which will influence future grid developments from our perspective.

**Need for more control.** As reflected by the current spectrum of available grid devices and mentioned by the participants, users wanted to gain more control during interaction. This includes expressiveness related features such as MIDI MPE abilities but also the combination with other UI elements which are more suitable for certain tasks.
**Protecting the white canvas.** Adding more controls stands in clear contrast with users’ wish to maintain the grid as a *white canvas*. A hardware platform that is more about enabling them to realize and implement their own musical ideas, then about offering a single predefined musical tool. Or rephrased, the power of generic grids is more to be potentially anything, than to be a specific *thing* which would imply restrictions regarding other approaches.

**Discussion**

While we found that grids theoretically have certain usability issues, our online survey indicates a very different picture. Users rejected these issues and pointed out that these can also represent advantages rather than shortcomings. While, these questions clearly polarized the participants, they all agreed on haptics being an advantages of grids over comparable interfaces on e.g. touchscreens. Here, we clearly see the advantage of *tangible properties* such as the ability to tactually experience objects [61] by feeling their surface, feeling distinct interaction elements [62], and experience a passive haptic feedback when pushing buttons [63].

This is a feature which is appreciated by musicians [64][65] when comparing physical instruments to musical touch screen applications in general. If the only advantage of grids is their haptics, there is a vivid discussion among the community that in the foreseeable future advanced display technologies, which incorporate shape changing surfaces [66][67][68], will make grid devices obsolete.

Our study points in a different direction. While we presented restrictions related to the low-res nature of grid devices to the participants, they insisted that these are creative constraints. By having the possibility space limited their *creative processes were fertilized* [69] and the users perspective on musical processes is constantly questioned and reinvented.

Looking at these constraints, we argue that they also add to the complexity of grids. The limited resolution prevents complex information to be displayed and binary input restricts interaction and expression. So why do users stick to such complex interfaces over other options? A derived hypothesis could be that musicians in general are used to high-threshold interfaces.

Learning to play instruments not only requires the development of physical abilities but also to create an understanding of the underlying context and processes. While these factors slow down the learning process they enable creative expression in the long run. As interaction designers working in the musical domain we not only have the
responsibility to design technologies that are usable for novice users [70], but we also have to work as NMI-luthiers [71] creating instruments which enable users to reach their full potential.

While “acoustic musical instruments have settled into canonical forms, taking centuries, if not millennia” [72], interfaces for NMIs only had decades to explore alternative forms of control so far. Such as the musical keyboard is not exclusive to pianos in the acoustic realm, the grid is the closest artifact we have to a ubiquitous canonical interface for NMIs.

Considering usage patterns and interactions we can already observe that specific approaches are more successful than others. How to think of musical layouts, how to represent time-based information, and how to interact is further converging to a more standardized way. Interestingly, the collected answers of the participants in our online survey did not clearly distinguish between the concept of a controller and an instrument, just as most pianists would not see the keyboard as a part that can be separated from the piano as a whole.

This more or less indicates that the grid for the users becomes the instrument. However, can the grid as a blank, flexible, and generic interface ever be a canonical instrument? Hence, instead of conceiving the grid as a canonical instrument, we should rather conceptualize it as a canonical interface form. While this interface form is currently restricted by design, users asked for more expression and control.

The question persists, how to offer these without compromising a grid’s genericness? We pointed out three yet unexplored design approaches to expand the capabilities of generic grid devices. We further showed through our data set that even if the grid turns into a common and widespread interface, further exploration and refinement is still to be done. We believe that the expressive power of touch gesture input can be further enhanced by integrating touch technology directly into the silicone, as was done by Teyssier et al. [73], making the silicone the matter implicitly involved in the sound generation process.

**Conclusion**

Proprietary hardware interfaces are closed systems by design, however, the monome design philosophy and the resultant community are pushing the idea of the grid as an open platform for musical interfaces. Hardware manufacturers are already implementing common protocols (MIDI, OSC) in their devices which enables users to hack, reuse, and customize these. In this vein, more manufacturers start to publish
documentations and offer frameworks to easily apply their interfaces beyond the intended context (*Novation, Ableton, Roli*). Even software solutions are offered for users without coding-knowledge to customize their devices using toolbox systems.

In summary, we consider the current developments as an exiting opportunity to further investigate new interaction concepts in the realm of grids which keep the generic qualities of the original idea and thus expand the possibilities users of grids have in the design of their applications without compromising on features such as form factor or *expressiveness*.

The collected works in this paper present an overview over the last 15 years of grid development since the introduction of the *Tenori-On* and the *monome grid*. We consider the here proposed ideas as a basis for discussion about the future of grid controllers. Overall, we see the grid as an increasingly important interface for NMIs that offers different approaches to tasks of modern musicians such as playing and sequencing instruments and furthermore due to its openness pushes users to delve into the interface and thus explore their individual art of interaction.

**Footnotes**

1. [https://www.britannica.com/art/keyboard-instrument#ref53738](https://www.britannica.com/art/keyboard-instrument#ref53738)


3. We will later use forum quotes to emphasize usability issues and opinions in the context of grid devices.

**Citations**


https://doi.org/10.1145/2984511.2984565


https://doi.org/10.1145/3430524.3440628

https://doi.org/10.1145/1979742.1979895

https://doi.org/10.1145/3290605.3300504

https://doi.org/10.1145/3242587.3242605

https://doi.org/10.1145/1518701.1518933

https://doi.org/10.1145/1978942.1979000


64. Altinsoy, M. E., & Merchel, S. (2018). Touchscreens and Musical Interaction. In S. Papetti & C. Saitis (Eds.) (pp. 239–255). Cham: Springer International Publishing. [https://doi.org/10.1007/978-3-319-58316-7_12](https://doi.org/10.1007/978-3-319-58316-7_12)


Baton Rouge, Louisiana: Association for Computing Machinery.
https://doi.org/10.1145/1226969.1227012


