GEM-NI: A System For Creating and Managing Alternatives In Generative Design

Loutfouz Zaman*, Wolfgang Stuerzlinger†, Christian Neugebauer‡, Robert Woodbury†, Maher Elkhaldi†, Naghmi Shireen†, Michael Terry§

*York University, Toronto, Canada, zaman@cse.yorku.ca, †Simon Fraser University, Vancouver, Canada, ‡University of Applied Sciences Bonn-Rhein-Sieg, Germany, §University of Waterloo, Canada

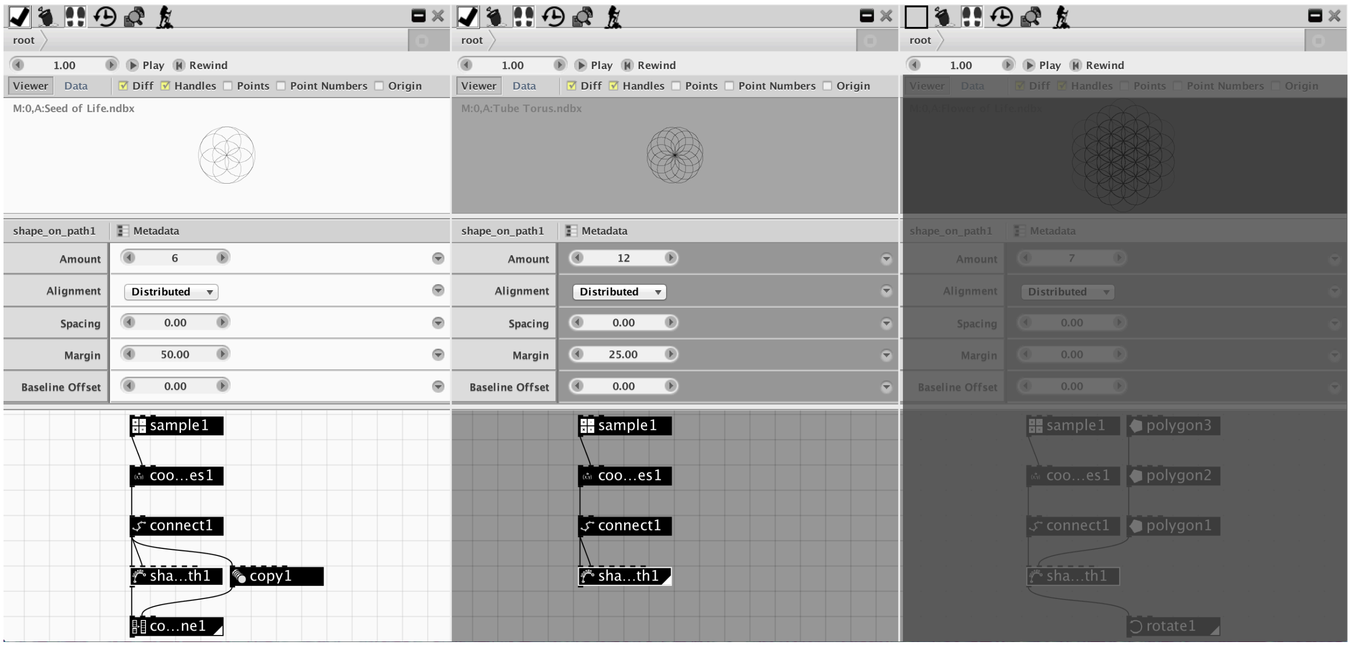

Figure 1. *GEM-NI* enables users to work with alternative generative designs simultaneously. Specifically, *GEM-NI* provides tools to manage the set of alternatives affected by edit operations, post-hoc merging of (parts of) alternatives, and several ways to create new alternatives, such as resurrection of past states with full undo lineage duplication or selection from an enhanced design gallery implementation. The leftmost alternative is the original design and is active, the center one is passive, and the rightmost one is idle.

ABSTRACT

We present *GEM-NI* – a graph-based generative-design tool that supports parallel exploration of alternative designs. Producing alternatives is a key feature of creative work, yet it is not strongly supported in most extant tools. *GEM-NI* enables various forms of exploration with alternatives such as parallel editing, recalling history, branching, merging, comparing, and Cartesian products of and for alternatives. Further, *GEM-NI* provides a modal graphical user interface and a design gallery, which both allow designers to control and manage their design exploration. We conducted an exploratory user study followed by in-depth one-on-one interviews with moderately and highly skills participants and obtained positive feedback for the system features, showing that *GEM-NI* supports creative design work well.

CHI 2015, April 18 - 23, 2015, Seoul, Republic of Korea

ACM 978-1-4503-3145-6/15/04…$15.00
http://dx.doi.org/10.1145/2702123.2702398

Author Keywords

Alternatives; generative design; parallel editing; graphical operation history; exploration; design gallery

ACM Classification Keywords

H.5.2 User Interfaces: Graphical user interfaces (GUI), I.3.4 [Graphics Utilities] – graphics editors

INTRODUCTION

Experts typically generate sets of alternative solutions when solving ill-defined problems [28]. This has been shown to result in higher quality outcomes [9]. For example, it is common practice for web designers [25], architects [1,24] and software engineers [31] to generate sketches of potential designs as they solve problems. These sketches help designers externalize knowledge, better understand the problem, and explore the space of potential solutions [2]. Due to the ease with which parameters can be varied, *parametric modeling*, where solutions are represented as models, is a particularly compelling technology to assist in exploring a design space. A *parametric* model represents the structure and hierarchy of a solution, the result of which is determined by relationships, constraints, and choices of parameter values at a given time. While building such models requires more effort, parametric modeling systems make variations easier and are now used in various domains. A prominent example is architectural design, as in the construction of the International Terminal Waterloo in London, England [37], pp. 43-45. Parametric models are often presented as networks of operations in a dataflow programming environment. As a given parametric model can quickly generate multiple options, the design process is often referred to as a *generative* one.

Parametric and generative design qualitatively changes the design process. The inherent capabilities of such systems also create new opportunities for design support tools. With generative design, it is possible to easily create a very large set of viable design options that satisfy a given set of constraints, as defined in the model. These alternatives represent points in a high-dimensional design space that must be explored, narrowed, and filtered. Current systems represent models as graphs, e.g., *AutoCAD DesignScript Studio, Grasshopper 3D*, *GenerativeComponents*, *Houdini, NodeBox,* and *Max/MSP*; or as trees, e.g., *CATIA, Inventor, ProE*, and *SpaceClaim*. Adoption is widespread in avant-garde practice and design schools and there are established textbooks and professional development courses.

In this paper, we present *GEM-NI*, a graph-based 2D design tool that supports the exploration of design alternatives in various ways. *GEM-NI* is built as a branch of *NodeBox*, a vector graphics generative design tool that uses graphs to express the computation behind the design. *NodeBox* has been used for visualization and generative art. Examples include visualizations of real-time data, evolutionary art installations that react to users, documents in a single visual style but with variations across pages, and customized wallpapers based on e-mail spam [29]. *GEM-NI* adds several novel features:

- interactive, selective post-hoc merging in alternatives;
- an enhanced interactive design gallery that explores ranges of parameters and *structural* changes to the model.

In the context of generative design, *GEM-NI* presents the following new features:

- methods to control parallel/linked editing: checkmarks and sandboxes;
- a non-destructive method for resurrecting past states from history with undo lineage, via enhanced "skating" [35];
- methods to quickly generate alternatives via branching;
- local and global undo;
- tools to manage alternatives and visually compare them;

To investigate our design choices we evaluated our system with moderately and highly skilled users.

We use the example in Figure 1 to demonstrate the capabilities of *GEM-NI*. Imagine that Ann, a designer, is tasked with creating a design for a book cover. To match the book content, she initially selects the "Seed of Life" pattern and recreates it in *GEM-NI*, by using three nodes (SAMPLE1, COORDNIATES1, CONNECT1) to create a circle. She then distributes copies of the circle along the same circular path (SHAPE_ON_PATH1), Figure 1 left. As the result does not seem complex enough, she goes back in the operation history and generates a new clone from an earlier state. After some parameter variation, she arrives at the "Tube Torus" design shown in Figure 1 center. Not entirely satisfied, she branches this design again to create a third, where she uses repeated polygons to arrange the circles in a more complex pattern, the "Flower of Life". Next, Ann creates several more alternatives using the Cartesian product (not shown in Figure 1) and uses individual and linked editing to tune the designs' parameters and manages them in the workspace. Throughout this, she uses local and global undo to correct mistakes. Merging is illustrated with another worked example below.

Encouraging exploration of a design space with parallel alternatives in design tools is a subject of current research. We are not aware of any mainstream end-user tools that yet support this approach to design exploration. Thus, we ran a user study with moderately and highly skilled users to investigate the appropriateness of our approach by gathering feedback and understanding the implications of parallel editing on their design process.

RELATED WORK

Extant Media for Exploration

Most exploration in the conceptual stage of design happens with the aid of pencil and paper—sketching is central. Its immediacy and speed enable a skilled designer to offload memory onto paper, thus greatly expanding the ability to make and consider design ideas. Buxton [5] suggests 11 qualities exhibited by (and possibly definitive of) sketching. Computer-based conceptual design systems often do not support these qualities adequately. Despite many methods proposed by researchers for the use of CAD in early stage conceptual design [17,36], CAD is still mostly used in the final stages of design, though this is changing [37].

We postulate that supporting design alternatives explicitly can capture some of the functionality seen in manual sketching, something almost entirely lacking in current CAD tools. Consider Ann, who: needs to shift focus quickly among her different designs, wants the capacity to represent, borrow, and reuse ideas across different alternatives, needs the ability to visually compare her solutions, and may even want to do parallel editing across multiple designs. *GEM-NI* provides such features.

Generative Design

We use the term *generative design* to label any computer-aided design system that provides tools to vary designs beyond direct manipulation of specific design elements.

Such systems can be understood as lying along a spectrum from direct manipulation to fully automatic design. Thus we consider parametric modeling, where changing parameters is a tool on top of manual model construction, to be a minimal generative system leaving most control with the designer. Genetic algorithms form an opposite and delegate exploration to the computer, while permitting user selection only at intermediary generations [39].

Here, we focus on the direct manipulation end of the spectrum, enabling designers to interactively control design decisions and provide "power tools" to make, modify, track, evaluate and visualize their work and to explore a larger number of design options. One of the key challenges is to facilitate the fluidity of the design process where many threads of possibilities are developed in parallel. Another challenge is a balance of features, user interactions, and workflows to enable designers to focus on exploring alternatives, as opposed to just managing them. *GEM-NI's* design gallery interface, described later, employs elements of genetic algorithms, but enhances them through Cartesian products of generative networks. The exploration facilities of *GEM-NI*, also described later, enable designers, such as Ann, to explore a much larger number of design options than what is possible through manual interaction.

Graphical Operation History

According to Shneiderman [27], history mechanisms can play an important part in the design process, supporting iterative analysis by enabling users to review, retrieve, and revisit visualization states. Many recent systems provide history-keeping mechanisms in the form of a timeline [19]. When interactive [11], graphical histories can amplify the exploration capabilities of a system. Users then can not only go back in time [13] and defer decisions [37, 38]; they have a mechanism to try out variations [35] by creating revisions [6] and versions [10] of the timeline. Moreover, history tools can help users to create reports or presentations, facilitating communication. Heer et al. [16] and Grossman et al. [13] together provide a comprehensive survey from an HCI perspective. Here we focus mainly on works that employ time sliders and graphical histories, as they are most relevant to histories in parametric design.

One of the earliest examples of graphical history is *Chimera* [20], which features an editable graphical history through panels depicting the results of each user operation. *Chronicle* [13] is a sophisticated system for exploring a document's history via a zoomable and track-based video playback metaphor. *Diffamation* [8]*, a* system for text version differencing employs a *time slider* that permits the user to explore differences over time. Su [32,33] presented new pictorial visualizations for the operation history of 2D vector illustrations as interactive storyboards.

GEM-NI's new resurrection feature permits Ann to resurrect past states non-destructively as new alternatives through a simple GUI that enables her to scroll through previews of past states or to select specific operations.

Design Space Exploration

Woodbury and Burrow [38] argue that design activity is well-modeled by a network structure. This network reflects the strategies and structure of the designer's exploration. They introduce the notion of *design hysteresis*, which points out that insights are discovered not just by explicitly visiting parts of a design space, but also through re-combination of visited alternatives in said *hysterical design space*. *Design Galleries* [23] explores this space by automatically generating and organizing variations of graphics or animations produced by a parametric model. Ma [22] introduces an interactive and dynamic graph representation of a database and argues that this type of visualization enhances the user experience in exploring the data. Jankun-Kelly and Ma [18] introduce a 2D spreadsheet visualization for multi-dimensional database exploration. Terry et al.'s *Parallel Pies* [35] enhance the user experience in generating and comparing alternatives by displaying several alternatives of a model with varying parameter values simultaneously. Lunzer and Hornbæk [21] present a novel subjunctive interface with multi-state sliders for each parameter to enable parallel exploration of a parametric space. They also show multiple states of a document side-by-side and permit users to rearrange these views.

Sheikholeslami [26] realized the *Dialer* interface in Bentley Systems' *GenerativeComponents* to enable interaction with the hysterical design space. The *Dialer* comprises concentric interactive rings, one for each parameter, where the ring divisions correspond to the explored values. The outermost ring illustrates the explored parts of the hysterical space. This compact visualization is limited in the number of divisions and parameters that can be visualized.

GEM-NI extends previous work by enabling users to select an arbitrary range of parameters and parts of a generative network for a design gallery. Together with the novel ability to explore structural products of generative networks this enables designers, such as Ann, to explore the design space more quickly and widely than with other approaches.

Interacting with Alternatives

Minimal support for alternatives is found in industry tools such as Autodesk's *Showcase* and Dassault Systèmes *SolidWorks* ([37], pp. 276-277). Both focus mainly on supporting alternatives through configuration management, *alternative lineup* features and side-by-side spreadsheet-like user interfaces. *CATIA* by Dassault Systèmes shows alternatives in *Catalogs*, a static gallery of assemblies and parts. It does not support interactive exploration.

Work on subjunctive interfaces [21] supports interaction for side-by-side exploratory analysis via viewing and editing of parametric models with a multi-handle slider user interface. Terry et al. [34] presented techniques to better support systems for parametric variations; *side views* – an on-demand command preview, the *parameter spectrum* – a replacement for the traditional slider control to display a range of possible results, and the *design horizon* – a complementary design space visualization. Heer et al. describe branching history as a way to remember operations that have been undone [16]. In contrast, Terry et al. [35] describe undo as a tool for reflection-in-action, in other words, for exploring variations. In Terry's *Parallel Paths*, when users duplicate a particular variation, its lineage is also duplicated, unlike in standard cloning. This copied history enables users to create variations *after* a command: when a result is unanticipated, but still worth keeping, users can duplicate the current state and then *non-destructively* return to a previous one. Terry et al. call this *skating*.

Juxtapose [15] presents a parallel code editor and runtime parameter tuning environment for GUI alternatives. When alternatives are linked, any block of code written in one alternative is shared among the rest. Working with this system requires strong coding skills. Bueno et al. [4] use the metaphor of *rewriting history* to enable users to manage variations and explorations of a design, with support for merging, generalizing and specializing. The *d.note* tool [14] also offers alternatives, together with revision control, change tracking and annotations. A user study on computational sketching tools compared three interaction models for working with alternatives in early design: a tab interface, a layered canvas, and spatial maps [30]. They found that spatial maps better support idea reflection, as they permit side-by-side comparisons.

GEM-NI enables interaction with multiple alternatives through parallel editing, history keeping, cloning, support for non-destructive resurrection, and new methods for easy management of alternatives. To enable designers to reuse their work more easily, *GEM-NI* adds a new method for post-hoc merging of (parts of) divergent alternatives.

GEM-NI

The name *GEM-NI, Generative Many-Nodes Interpreter*, is inspired by the many-worlds interpretation in quantum physics. It implies that all possible alternative histories and futures are real, each representing an actual "world". We focused on 2D graphics, a domain that offers sufficient complexity for common issues and patterns in generative modeling to emerge, yet still practical for user studies. Also, *GEM-NI* supports exploratory design tasks widely used in the design literature and in HCI in Green's cognitive dimensions of notations [12].

We already introduced a simple usage scenario for *GEM-NI* above. Beyond this, *GEM-NI* is capable of handling a large number of alternatives, the exploration of which is only limited by screen size, processing power and memory. Each alternative is hosted in a panel. Panels are contained in one or more *workspace(s)*, which can be saved. Figure 1 shows such a workspace with three alternatives. Multiple workspaces may be open at the same time. The panel for each alternative consists of three views: output, parameter, and network view. To facilitate side-by-side viewing of alternatives, *GEM-NI's* panel layout differs from that of *NodeBox*. In *GEM-NI* views are stacked vertically: output on top, parameters in the middle, and network view at the bottom. The order in which alternatives show on screen can be re-arranged by drag and drop using a modifier key, with preview and target location highlighting. In the conceptual design phase, designers routinely generate dozens of sketches. That amount of content is difficult to fit onto a single monitor, if all alternatives are still to be viewable and editable. To aid the designer in keeping an overview of all considered alternatives, to organize them, and to view them side-by-side for visual comparison, we support multi-monitor setups in *GEM-NI*. In a preferences menu, the user can select from 1×1 to (currently) 2×3 monitors. The workspace is then re-arranged to spread all alternatives as evenly as possible for the chosen monitor arrangement. Within each monitor, horizontal space is evenly distributed.

In *GEM-NI* the creation of nodes, their positions, parameter values as well as selection state are synchronized by default across all editable alternatives. Thus, moving a node or changing a node parameter affects all of its instances in other editable alternatives. Such parallel editing can be enabled or disabled, see below. We found that uniformity in network layout makes it easier for designers to identify common elements and to compare networks visually across alternatives. E.g., SHAPE_ON_PATH1 in Figure 1 was selected in the leftmost alternative and is now selected everywhere with the corresponding parameter views. For the same reasons, zooming and panning on the network and output view are also synchronized. Every operation is accessible through the menu bar. Important ones are also accessible through GUI buttons or keyboard shortcuts.

Parallel Editing

The most common use case for parallel editing is parameter variation in a design. Ann might use parallel editing to change the size of multiple alternatives, which saves having to repeat the operation in each one. Or she might add a new node that adds a background rectangle to all designs. In *GEM-NI* she can control which alternatives are *idle*, i.e., non-editable, through *checkmarks*. Selecting an alternative makes it *active*. All other checked alternatives are *passive* and thus subject to parallel edits. Often, Ann wants to focus only on a single alternative, i.e., work in a *sandbox*.

A workspace typically contains multiple passive and idle alternatives. The active one, Figure 1 left, is shown in bright gray. Passive alternatives, Figure 1 middle, are shown in a mid-tone gray, and idle (unchecked) ones are dark gray, Figure 1 right. An alternative is activated simply by clicking anywhere on its panel, or by switching to it via the TAB key. Newly created alternatives are set to passive, permitting parallel editing. Editing an alternative makes it active and pushes changes to all passive alternatives. Idle alternatives remain unchanged. Pushed changes include operations in the network view (e.g., creating, renaming, deleting, connecting or disconnecting a node), in the parameter view (e.g., tweaking a parameter of the node), and in the output view (e.g., moving or resizing a shape by direct manipulation). Sandboxing addresses the case when the designer wants to focus her edits on only a single alternative (Figure 2). This functionality simply idles (unchecks) all other alternatives. Both checkmarks and sandboxing are accessible through GUI buttons (✔ and [icon], see Figure 2) or through modifier keys when clicking on an alternative. The “mute” and “solo” buttons in audio and video software, such as Adobe *Audition CC* and Apple *Final Cut Pro X*, inspired our checkmarks and sandboxing.

Local and Global Undo

GEM-NI supports two types of undo: local and global. Local undo refers to undo in the currently active alternative. Global undo undoes in all checked alternatives in the workspace. Performing local undo or redo clears the global undo stack to avoid undo synchronization problems. A more powerful undo system, e.g., [11], could address this limitation. We did not implement this for simplicity.

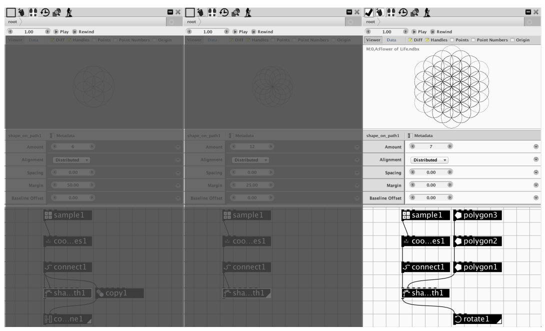

Figure 2. The alternative on the right is sandboxed. The first two alternatives are therefore idle.

Selective Merging

Designers frequently branch out to explore different alternatives. Sometimes they then want to re-use new parts in other alternatives. Figure 3 illustrates a merging scenario in *GEM-NI*. Inspired by Brownian motion, Ann first created a grid with randomly displaced ellipses (Figure 3a left). She then created an alternative that uses a compound of an ellipse and a circle (center). Subsequently she created a slightly more structured 10×10 grid of compound circles (right). Looking at this, she likes the result of the GRID1, WIGGLE1, and ELLIPSE1 nodes in the right design as well as the capability of varying the size of the grid through the new NUMBER1 node. Thus, she merges these four nodes into the other two alternatives. This overwrites the parameters of existing nodes and creates the NUMBER1 node with connections to GRID1 in the other two alternatives. Note that a copy operation would not recreate these specific connections. She then changes the size of the grid in all three alternatives to 15×15 (Figure 3b) with parallel editing.

Selective merging is a new mechanism to ensure that parts of a design can be *post-hoc* integrated into other alternatives. *GEM-NI* implements this by *overwriting* the state of the *selected* nodes across all passive alternatives. This is different from standard copy & paste, which will *duplicate* existing nodes. When merging, nodes that do not exist are created and connected suitably in the passive alternatives. Parameters of common nodes are overwritten from the active alternative. This may create conflicts, which the user needs to address later. Performing merging on the complete network essentially turns all alternatives into clones (with potentially different undo history stacks). Our technique is inspired by the corresponding functionality in source code management, mainly *Git*.

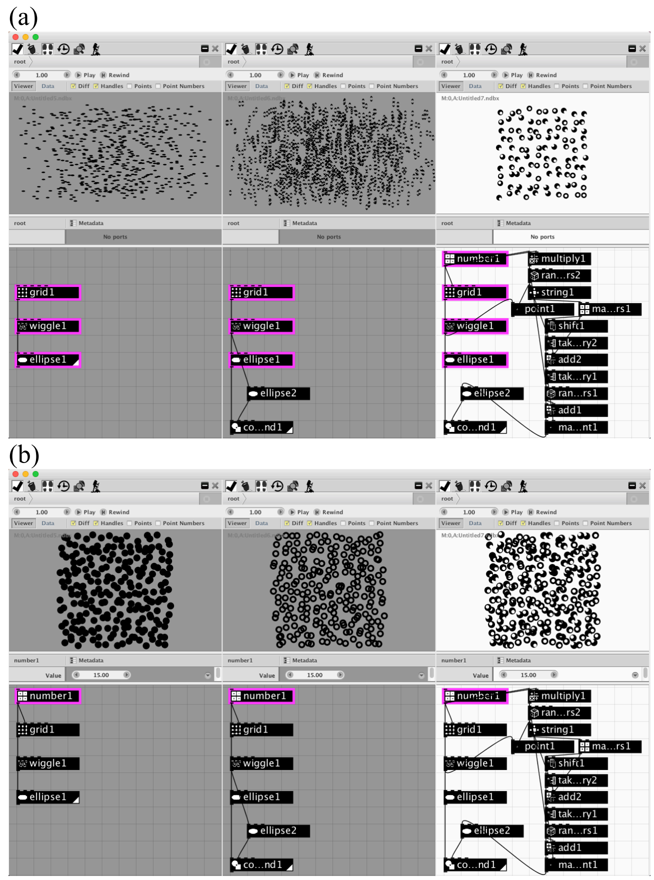

Figure 3. Merging and parallel editing: (a) Initial state with highlighted nodes selected for merging. (b) Merging replicates new nodes and connections into all other alternatives and overwrites parameters of existing nodes. The user then globally changes NUMBER1 to 15.

Creating Alternatives

Creating new empty alternatives is standard functionality. During branching/cloning, and in contrast to most other work, *GEM-NI* preserves the undo stack, which enables Ann to undo operations in both alternatives. In the example in Figure 1, Ann branched an alternative design from an intermediate version of her initial one. Usually this meant either relying on intermediate saves or using (destructive) undo to go back in time. In *GEM-NI* she can use the resurrection dialog to scroll back in time and select a starting point for a new exploration. Also, the enhanced interactive design gallery in *GEM-NI* enables Ann not only to explore the parametric design space but also to create new variations for the *structure* of the network. This makes it easier for Ann to quickly explore a larger set of designs.

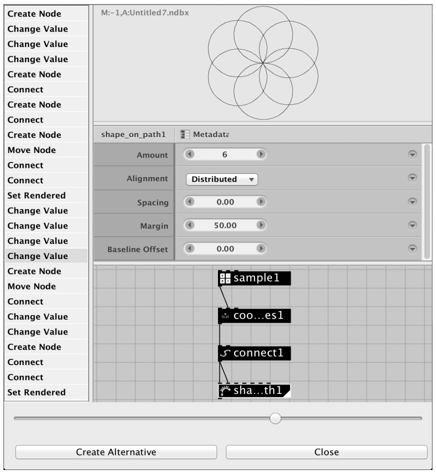

Figure 4. Dialog for creating an alternative from history: the history list on the left and the state of the alternative at that time on the right. The current entry is highlighted. The state can be selected from the list directly or by dragging the slider.

Branches

An alternative can be created as a branch through cloning, where the entire network from the active alternative is copied to the newly created alternative, along with its undo lineage. This can be interpreted as an adaptation of *skating* [35] to generative design. The new alternative then appears to the right of the active one. Preserving the undo stack upon cloning enables new use cases, as the user can now undo operations in both alternatives.

Resurrection from History

Creating alternatives from history, i.e., resurrecting past states, enables Ann to look through her past work and to select particular points in time from which she intends to "branch out" and to explore new alternatives. This happens in *GEM-NI* through a dialog (Figure 4). There, all states from the undo history are listed on the left, with a time slider on the bottom. Clicking on a list item or scrolling shows a preview of the corresponding state on the right. Clicking the "Create Alternative" button then instantiates the selected state as a new alternative in the workspace next to the active alternative. In Figure 4, the past state in the history of the left design of Figure 1 is highlighted, from which the second alternative was branched out. The dialog enables the user to create more than one alternative at a time. As with branching, *GEM-NI* clones the history stack on a resurrection from history. Together with the history previews this provides an enhanced form of *skating* [35].

Design Gallery

We implemented an interface for creating alternatives from a design gallery, inspired by the parametric Cartesian product in *Dialer* [26]. Our interface extends *Dialer* in several ways, most importantly by supporting structural products. Figure 5 shows a design gallery for Figure 1. Starting with the scenario in Figure 1, Ann now wants to explore the design space more widely. Going beyond *Dialer* [26], *GEM-NI* also enables her to specify the range over which parameters should vary. For this, Ann first selects (with a modifier key) two or more alternatives as the basis for the Cartesian product. This outlines these alternatives with a red frame. In the example in Figure 5 all three alternatives from Figure 1 are included in the product. Upon modifier key release, the main dialog appears (Figure 5a), which shows Ann all nodes of the alternatives in a nested list, with the second level denoting the parameters. To streamline the workflow, only common nodes whose parameters differ between the selected alternatives are selected for the product by default and expanded. SHAPE_ON_PATH1 is the only common node with different parameters (*Amount* and *Margin*) in the three alternatives in Figure 1. The GUI elements then permit Ann to include or exclude nodes and to specify ranges for the values of the nodes in the Cartesian product (Figure 5a). For ranges she can configure the minimum, maximum and step size.

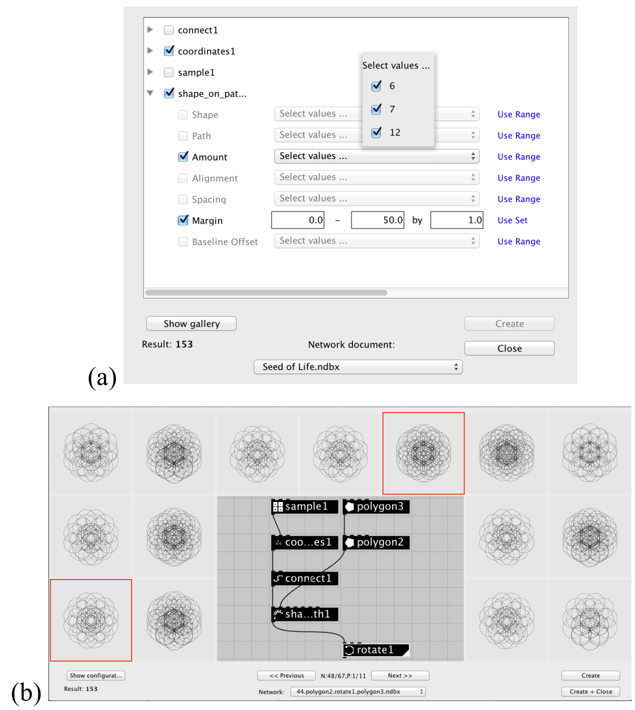

Figure 5. Design Gallery. (a) Cartesian product menu; (b) parameter range exploration preview of the selected node.

To give Ann an idea of how many potential results to expect, *GEM-NI* shows the number of designs that would be generated at the bottom left. When satisfied with the settings, Ann presses the "Show gallery" button, which displays the results of the Cartesian product (Figure 5b) in a dialog that enables the user to scroll through all pages of results with the "Previous" and "Next" buttons. She then selects two previewed designs by clicking on them and presses the "Create" button to instantiate the selected designs as new alternatives, which appends them as the last alternatives on the right side of the workspace. For reference, the design gallery shows the network of the first selected alternative in the middle, but enables Ann also to view other selected networks. Optionally, the *GEM-NI* design gallery can be invoked on a single alternative. In this case the gallery will operate in range only mode.

Finally, Ann can explore even more potential designs by exploring different *structures* for the networks of the generative model. For this our new Cartesian product first identifies all nodes that are different in the alternatives. Then *GEM-NI* constructs their power set and substitutes each into the network common to the alternatives. Since there are a total of 6 node differences among the three alternatives in Figure 1, $2^6 = 64$ networks are created. Ann can then scroll through all designs generated by varying both the network as well as the parameters or select a specific network from a drop-down list. E.g., in Figure 5b, she has selected a generated network that consists of three substituted nodes: POLYGON2, POLYGON3, and ROTATE1. In the example shown in Figure 5 there are 64 generated networks with 153 parameter variations. Thus this design gallery contains over ten thousand potential alternatives.

EVALUATION

To confirm the appropriateness of the presented techniques for the design process, we evaluated *GEM-NI* in two steps: a workshop with an exploratory design study, and follow-up in-depth interviews. The workshop introduced *NodeBox* and had the goal to gather feedback on *GEM-NI* from a user group moderately experienced in generative design.

Participants

Five unpaid participants (2 female) were recruited through a "session on new generative design tools" announcement. We targeted participants experienced with generative design. Coffee and cookies were offered as incentive. Participants design backgrounds were: architectural, sound, visual, and information design, as well as arts. One participant had to withdraw during the workshop due to an appointment. The participants were between 21 and 31 years old (μ=27.2). All were experienced designers (μ=5.7 years). One participant reported experience with generative design tools, namely *Grasshopper*, another knew *Processing*. None knew *NodeBox*. In the pre-questionnaire, all reported that they routinely create multiple alternative designs as opposed to a single solution. All stated that when they design, they regularly keep track of, review, and revisit their design iterations. For that, participants reported the following methods: saving multiple files (even for minor changes), creating files from scratch for major conceptual changes, using a stylus with a note taking application, various combinations of shuffling between files, sketches, tracing paper, and images, and keeping everything in a notebook and/or printouts, including intermediate artifacts.

Apparatus and Procedure

Workstations with dual monitors running *Windows 7* were used. Blank sketch sheets and pens were supplied. The workshop was split into two phases, followed by a one-on-one in-depth interview with participants at a later date. Along with logging and interviews, we used the Creativity Support Index (CSI) [7], a quantitative, psychometric tool in form of a survey to assess how well both systems assist creativity in the design process. In both phases participants completed the CSI questionnaire after the creative task. Collaboration was not rated.

The Workshop

In the first workshop phase participants were taught a basic version of *GEM-NI,* feature-wise equivalent to *NodeBox,* i.e., without our new contributions, and called *NodeBox* for the remainder of this section. We picked examples from a generative design book [3] as tasks. We first demonstrated how to create six designs and asked participants to recreate them on their workstations. This familiarized participants with the system to enable them to perform the main creative task. Said task was a design scenario, where participants had a client that wanted them to come up with an "algorithmic shapes" design for a small front door window, about 20×20 cm. The client expected options to choose from. Participants then used *NodeBox* to create a number of alternative designs and saved them as different files. Participants were free to sketch with pen and paper, as necessary, and were given 30 minutes for the task.

In the second phase, participants first learned how to use *GEM-NI*. All above-presented features of the system were demonstrated on the example in Figure 1, which they recreated from scratch. The second task stipulated that their client saw the alternative designs for the door window and liked them so much that she asked them to extend them to cover the entire door. Participants could reuse their designs from the first task. They were given 30 minutes to complete the task and again permitted to sketch. After completing the second CSI questionnaire they also filled the paired comparison part of the CSI. Since the tasks in our study were similar enough, we performed the CSI evaluation as a within subject tool comparison with the same task.

Results of the Workshop

The small number of participants limits our results. Also, we evaluated the tools in a fixed order. Yet, counterbalancing is difficult, as *GEM-NI* is based on *NodeBox.* Looking at the first phase, we noted that several participants had already created alternatives in a *single document* in *NodeBox* (without prompting!) through the ability to create multiple output nodes. This likely reduces the "cool tool" bias and verifies that designers already *plan for alternatives*, even in current tools. These participants had primed themselves to learn *GEM-NI* quickly. Two participants, P1 and P4 created four alternatives with *GEM-NI*. All designs of P1 differed only in parameters, but not structurally. P4 came up with several designs that were structurally different. The logs showed that both P1 and P4 used branching to create their alternatives and worked non-linearly, i.e., went back and forth between alternatives. We also logged five instances where P1 rearranged the order of alternatives. Another participant, P3, created two alternatives and accessed the history and design gallery, but did not create alternatives with those methods. Only one person was observed to use pen and paper briefly.

In-Depth Interviews

We ran in-depth interviews with three of the participants (P1-3) in the days after the workshop. At the start of the interview, participants were given a short review of the features of *NodeBox* and *GEM-NI*. Then they were asked to continue working on the second task with *GEM-NI*. In a variant of a think-aloud protocol, we asked participants to express their opinions during this, to make comments on the tools, to provide feedback on the overall workflow and experience, and to explain why they made their decisions. P1 and P2 completed the interview in a little over two hours. Participant P3 was only able to dedicate 30 minutes. Overall, participants used many more features of *GEM-NI*.

P1 created two alternatives from history. When asked about this feature, P1 pointed out "*creating alternatives from history is superior because I like the idea of being able to pick something from the actual history, which could contain ideas that were not further developed*". This is in contrast to the alternative workflow of branching and deleting of unwanted parts of the graph, where he added "*[with this] some steps might not be captured, such as creation and deletion of connectors and partial editing of nodes*". During the interview, P1 produced designs that were different both structurally and in parameters. In the end, P1 created seven designs in a non-linear way through branching. He also created an alternative from a design gallery and two from scratch. We logged deletion of six alternatives. He re-ordered alternatives 30 times and used global undo once. P2 created five designs in two workspaces and saved some of his alternatives individually and then opened them in a new workspace. He re-ordered alternatives seven times, created two alternatives from scratch and five non-linearly as branches and used merging. P2 deleted six alternatives. P2 was not able to leverage design galleries, as his design had very subtle variations in only two parameters and thus "*the results shown in the gallery were [almost] identical*". P3 created five alternatives in a single workspace in a short time, through a design gallery, which P3 found to be "*a great way to explore possibilities*". P3 deleted one alternative and rearranged alternatives 21 times.

All participants were observed using the two available monitors. It was important to their workflow to focus on a single alternative. Therefore, all dragged idle and passive alternatives to the secondary monitor, to increase the workspace for the active alternative. This generated many instances of rearranging. P1 and P2 had programming skills and stated: "*GEM-NI is like version control [systems]*". They drew on their experience with *Git* and used *GEM-NI* somewhat like a version control system for design, which enabled them to experiment more. P3 was not familiar with software version control and thus did not have the corresponding mental model. P2 demonstrated an unexpected use of our system. He created multiple sub-graphs as alternatives, where the output of all sub-graphs was rendered inside a single *GEM-NI* alternative. Then, he started using the panels as means to explore even further alternatives. In the freeform feedback, he later wrote: "*I encountered some unexpected designs while using [GEM-NI], which made things much more interesting than I had first imagined. Interactive [parallel] editing had very interesting results*". P2 worked always on a single alternative at a time and perceived parallel editing of multiple alternatives at the same time "*to be hard*", likely due to the increased cognitive effort required for such parallel tasks. This corresponds to the experience with *Juxtapose* [15], which requires strong coding skills. Only one participant was observed to use global undo.

All participants complained about aspects of *NodeBox* and to a lesser degree about *GEM-NI's* features. Most criticism revolved around the fact that focusing is not automated enough. They found rearranging panels to be hard and wished for an easier workflow. We did not focus on streamlining this specific task. Participants also asked for some difference visualization that "*would highlight changes in the rendered geometry and network*". They also preferred that sandboxing be the default work mode, and that parallel editing only be available on demand (opposite to our default). Finally, participants P2 and P3 wanted to minimize or collapse alternatives, instead of being confronted with all of them simultaneously. They suggested a side window or panel that "*shows alternatives in a way similar to the design gallery*". They also identified that they would like alternatives that are collapsed to automatically turn idle.

Task	Factor \ Scale	Enjoyment	Exploration	Expressiveness	Immersion	Results worth effort
1	Factor counts (σ)	2.5 (1.7)	3.8 (1.5)	4.0 (0.8)	2.3 (1.3)	2.5 (1.3)
1	Factor score (σ)	13 (2.5)	11.3 (2.5)	9.0 (4.3)	6.5 (1.3)	11.5 (2.4)
1	Weighted factor score (σ)	34.8 (30.6)	41.0 (15.5)	36.5 (18.4)	15.3 (9.6)	27.0 (11.2)
2	Factor counts (σ)	2.5 (1.7)	3.8 (1.5)	4.0 (0.8)	2.3 (1.3)	2.5 (1.3)
2	Factor score (σ)	14.8 (2.8)	15.8 (1.5)	14.0 (3.2)	10.0 (7.1)	13.3 (5.0)
2	Weighted factor score (σ)	37.0 (28.9)	59.8 (25.5)	56.8 (20.5)	28.8 (34.8)	32.3 (20.1)

Table 1. Average results from first task using *NodeBox (top)*; second task using *GEM-NI (bottom)*.

Results of the CSI Questionnaire and Discussion

The CSI analysis, revealed an average score of 51.5 ($\sigma = 13.5$) in the first task and a substantially higher score, 71.5 ($\sigma = 12.84$), for the one where *GEM-NI* was used ($N = 4$ for both tasks). These results depend to some degree on individual's preferences and their level of expertise with the tool. Similar to [7], we report the results with respect to average factor counts, factor score and weighted factor score (Table 1). Average counts express the number of times that participants chose a particular factor as important to the task. Expressiveness and exploration were ranked as most important. The factor score sums both agreement statement responses for a factor. A higher number indicates better supports. Weighted factor scores are more sensitive to the factors that are the most important ones for the given task. In both dimensions, *GEM-NI* scored again much higher for expressiveness and exploration.

OVERALL DISCUSSION

Here we discuss some of the consequences of the design decisions behind *GEM-NI*, as backed by the outcomes of our workshop and the in-depth interviews. *GEM-NI* supports a number of ways to create alternatives. We include adapted and enhanced variations of results from previous work, such as *Parallel Pies* [35]. We also introduced a new method for creating alternatives from a graphical history with support for lineage duplication, i.e., graphical skating, for generative design tools. The in-depth interviews indicate that this is an ideal feature to easily explore what-if scenarios. Another noteworthy way of creating alternatives in *GEM-NI* is the design gallery. This interface improves previous work [26], by enabling users to select an arbitrary range of parameters and/or parts of the generative network to use. Furthermore, we added the new ability to create alternatives from the product of generative networks, which we believe to be a great addition to design space exploration tools and found to be useful in our evaluation. Notably, participants stated that "*[they] arrived at designs that they did not expect or foresee directly*". They attributed this to both the design gallery and to parallel editing. To control the scope of parallel editing, *GEM-NI* provides checkmarks and sandboxing, which participants found very useful during our evaluation.

In *GEM-NI*, we also introduced a novel method for post-hoc merging of alternatives, inspired by branch merging in *Git*. With post-hoc merging, a designer can easily "import" the knowledge embedded in a sub-network into another alternative. Post-hoc merging is particularly useful when a designer does not remember how he/she arrived at a particular state or if someone else modified the design. Yet, we only observed and logged one participant using the technique, potentially due to the limited design complexity explored in our evaluation. Still, we believe that with time, users will realize the full potential of this feature, similar to its pervasiveness in software projects.

Two participants rearranged alternatives extensively to focus on a single one on one monitor. This justifies our decision to support multi-monitors and to use such systems in the evaluation. Participants requested minimizing and other methods to manage alternatives on dual monitors. Participants also requested difference visualizations for alternatives, and overlaid history steps. Such features have been advocated before [21]. *GEM-NI* was implemented as a branch of *NodeBox 3* by adding multiple-document model support via universal unique identifiers. This enables consistent relationships between alternatives to persist even when they are kept offline. Alternatives can then safely be included back into the workspace at a later stage, without naming conflicts. *GEM-NI* supports versioning in this way and P2 used this during the evaluation.

While the sample size of our evaluation is small, we believe it to be representative for what moderately experienced designers can achieve with *GEM-NI* compared to traditional solutions, in terms of better exploration of a design space and expressiveness. The fact that participants even created alternative-supporting schemes in existing tools, underlines the need for *GEM-NI*'s approach in generative design tools.

CONCLUSION

We presented *GEM-NI*—a new system for creating and managing alternatives in generative design. The system supports parallel editing via checkmarks and sandboxing, two new methods to control which alternatives are affected by a parallel edit. Also, we introduced a novel method for post-hoc merging of alternatives. Moreover, *GEM-NI* provides several methods to create alternatives, including a new method for resurrecting alternatives from a graphical history with previews, with full lineage preservation. Another way to create alternatives is a new design gallery, which enables users to select which ranges of parameters and/or parts of the generative network model to use for exploration. Moreover, our design gallery supports a new method to explore products of generative networks.

The feedback from participants in a workshop and in-depth interviews suggest that *GEM-NI,* and more broadly the approach behind it, indeed enables designers to work more creatively. The results indicate the direct applicability of the presented techniques for the design process also via the CSI questionnaire. While the sample size of our user study is small, it identified the potential for better creativity support through alternatives in design tools.

Future Work

We envision several extensions. Currently, the network layout is not kept consistent across versions if merging or other editing occurs. We are also planning to incorporate difference visualizations and the ability to enlarge the active alternative, and to minimize others. We will filter visually similar candidates in the design gallery. Finally, we intend to evaluate *GEM-NI* on a 2×3 multi-monitor setup.

ACKNOWLEDGEMENTS

Thanks to the helpers for the workshop and to the GRAND NCE and NSERC Discovery program for funding.

REFERENCES

1. Akin, Ö. How do architects design? *Artificial Intelligence and Pattern Recognition in Computer Aided Design*, (1978), 65–103.
2. Akin, Ö. Variants in design cognition. *Design knowing and learning: Cognition in design education*, (2001), 105–124.
3. Bohnacker, H. *Generative Design: Visualize, Program, and Create with Processing.* Princeton Architectural Press, New York, 2012.

4. Bueno, C., Crossland, S., Lutteroth, C., and Weber, G. Rewriting History: More Power to Creative People. *OzCHI 2011*, 62–71.
5. Buxton, B. *Sketching User Experiences: Getting the Design Right and the Right Design.* Morgan & Kaufmann, 2007.
6. Chen, H.-T., Wei, L.-Y., and Chang, C.-F. Nonlinear revision control for images. *ACM SIGGRAPH 2011 papers*, ACM (2011), 105:1–105:10.
7. Cherry, E. and Latulipe, C. Quantifying the Creativity Support of Digital Tools Through the Creativity Support Index. *TOCHI 2014 21*, 4 (2014), 21:1–21:25.
8. Chevalier, F., Dragicevic, P., Bezerianos, A., and Fekete, J.-D. Using text animated transitions to support navigation in document histories. *CHI 2010*, ACM (2010), 683–692.
9. Dow, S., Fortuna, J., Schwartz, D., Altringer, B., et al. Prototyping Dynamics: Sharing Multiple Designs Improves Exploration, Group Rapport, and Results. *SIGCHI' 11*, ACM (2011), 2807–2816.
10. Drucker, S.M., Petschnigg, G., and Agrawala, M. Comparing and managing multiple versions of slide presentations. *UIST 2006*, ACM (2006), 47–56.
11. Edwards, W.K., Igarashi, T., LaMarca, A., and Mynatt, E.D. A temporal model for multi-level undo and redo. *UIST 2000*, ACM (2000), 31–40.
12. Green, T.R. Cognitive dimensions of notations. *People and Computers V*, (1989), 443–460.
13. Grossman, T., Matejka, J., et al. Chronicle: capture, exploration, and playback of document workflow histories. *UIST 2010*, ACM (2010), 143–152.
14. Hartmann, B., Follmer, S., Ricciardi, A., Cardenas, T., and Klemmer, S.R. d.note: revising user interfaces through change tracking, annotations, and alternatives. *CHI 2010*, ACM (2010), 493–502.
15. Hartmann, B., Yu, L., Allison, A., Yang, Y., and Klemmer, S.R. Design as exploration: creating interface alternatives through parallel authoring and runtime tuning. *UIST 2008*, ACM (2008), 91–100.
16. Heer, J., Mackinlay, J., Stolte, C., and Agrawala, M. Graphical histories for visualization: supporting analysis, communication, and evaluation. *TVCG 14*, 6 (2008), 1189–1196.
17. Hsu, W. and Woon, I.M.Y. Current research in the conceptual design of mechanical products. *Computer-Aided Design 30*, 5 (1998), 377 – 389.
18. Jankun-Kelly, T.J. and Ma, K.-L. A spreadsheet interface for visualization exploration. *Visualization '00*, IEEE Computer Society Press (2000), 69–76.
19. Klemmer, S.R., Thomsen, M., Phelps-Goodman, E., Lee, R., and Landay, J.A. Where do web sites come from?: Capturing and interacting with design history. *CHI 2002*, ACM (2002), 1–8.
20. Kurlander, D. and Feiner, S. A visual language for browsing, undoing, and redoing graphical interface commands. In *Visual Languages and Visual Programming*, 1990, 257–275.
21. Lunzer, A. and Hornbæk, K. Subjunctive Interfaces: Extending Applications to Support Parallel Setup, Viewing and Control of Alternative Scenarios. *ACM TOCHI 14*, 4 (2008), 17:1–17:44.
22. Ma, K.-L. Image graphs - a novel approach to visual data exploration. *Visualization '99*, (1999), 81–88.
23. Marks, J., et al. Design galleries: a general approach to setting parameters for computer graphics and animation. *SIGGRAPH '97*, ACM (1997), 389–400.
24. Menezes, A. and Lawson, B. How designers perceive sketches. *Design Studies 27*, 5 (2006), 571–585.
25. Newman, M.W. and Landay, J.A. Sitemaps, Storyboards, and Specifications: A Sketch of Web Site Design Practice. *DIS 2000*, ACM (2000), 263–274.
26. Sheikholeslami, M. You can get more than you make. 2011. http://summit.sfu.ca/item/9649.
27. Shneiderman, B. The eyes have it: a task by data type taxonomy for information visualizations. *Visual Languages 1996*, IEEE (1996), 336–343.
28. Simon, H.A. *The sciences of the artificial.* MIT press, 1996.
29. Smedt, T.D., Lechat, L., and Daelemans, W. Generative Art Inspired by Nature, Using NodeBox. In *Applications of Evolutionary Computation*. Springer Berlin Heidelberg, 2011, 264–272.
30. Smith, B.N., Xu, A., and Bailey, B.P. Improving interaction models for generating and managing alternative ideas during early design work. *Graphics Interface 2010*, (2010), 121–128.
31. Stephan, M. and Cordy, J.R. A survey of model comparison approaches and applications. *Conference on Model-Driven Engineering and Software Development*, (2013), 265–277.
32. Su, S.L. Visualizing, editing, and inferring structure in 2D graphics. *Adjunct UIST 2007*, ACM (2007), 29–32.
33. Su, S.L., Paris, S., Aliaga, F., Scull, C., Johnson, S., and Durand, F. *Interactive Visual Histories for Vector Graphics.* MIT, 2009.
34. Terry, M. and Mynatt, E.D. Recognizing creative needs in user interface design. *Creativity and Cognition 2002*, ACM (2002), 38–44.
35. Terry, M., Mynatt, E.D., Nakakoji, K., and Yamamoto, Y. Variation in element and action: supporting simultaneous development of alternative solutions. *CHI 2004*, ACM (2004), 711–718.
36. Wang, L., Shen, W., Xie, H., Neelamkavil, J., and Pardasani, A. Collaborative conceptual design—state of the art and future trends. *Computer-Aided Design 34*, 13 (2002), 981 – 996.
37. Woodbury, R. *Elements of Parametric Design.* Routledge, 2010.
38. Woodbury, R.F. and Burrow, A.L. Whither design space? *Artif. Intell. Eng. Des. Anal. Manuf. 20*, 2 (2006), 63–82.
39. Xu, K., Zhang, H., Cohen-Or, D., and Chen, B. Fit and diverse: set evolution for inspiring 3D shape galleries. *ACM TOG*, *31,* 4, (2012), 57:1–57:10.