applied sciences

MDPI

*Article*

# The Application of Generative Algorithms in Human-Centered Product Development

**Lewis Urquhart [1,*], Andrew Wodehouse [1], Brian Loudon [2] and Craig Fingland [1]**

[1] Department of Design Manufacturing and Engineering Management, Faculty of Engineering, University of Strathclyde, 16 Richmond St, Glasgow G1 1XQ, UK; andrew.wodehouse@strath.ac.uk (A.W.); craig.fingland.2016@uni.strath.ac.uk (C.F.)
[2] Loud1Design (Independent Design Consultancy), 234 West George Street, Glasgow G2 4QY, UK; brian@loud1design.co.uk
* Correspondence: lewis.urquhart@strath.ac.uk

**Abstract:** Algorithmic design harnesses the power of computation to generate a form based on input data and rules. In the product design setting, a major advantage afforded by this approach is the ability to automate the customization of design variations in accordance with the requirements of individual users. The background knowledge, intuition, and critical judgement of the designer are still essential but are focused on different areas of the design process. Thus far, little research has been applied directly to the problem of ergonomics in generative design. In this paper, we review the relevant literature in generative design, topology optimization, and computational design in order to describe the ways in which algorithms can be incorporated into the design process from a human-factors perspective–design tailored around human anatomy and usability requirements. We then develop a model for approaching generative design development work, oriented around human factors (particular ergonomics), and describe a case study from the PRIME-VR2 research project in which an algorithmic workflow utilized user scan data and 3D-printing technology to generate bespoke versions of a standard controller device.

**Keywords:** generative design; algorithmic design; human-centered design; human-factors; ergonomics

check for updates

**Citation:** Urquhart, L.; Wodehouse, A.; Loudon, B.; Fingland, C. The Application of Generative Algorithms in Human-Centered Product Development. *Appl. Sci.* **2022**, *12*, 3682. https://doi.org/10.3390/app12073682

Academic Editors: Byung Yong Jeong, Ken Nah and Yonghwan Pan

Received: 9 March 2022
Accepted: 4 April 2022
Published: 6 April 2022



## 1. Introduction

Algorithmic design harnesses the power of computation to explore a greater diversity of concepts around a particular design goal in the development process. The background knowledge, intuition, and critical judgement of the designer are still essential but are focused on different areas of the design process. This includes developing the basic abstraction of the problem, designing algorithms for the basic form and constraints, the selection of promising avenues of exploration, and the refinement of problem parameters. These activities require the same creativity, intuition, and judgement normally associated with innovative design working. The generation of the algorithms and code requires logical means of thinking and skills in software that may not be currently familiar to designers.

While the uptake of generative design methods is fairly limited in professional design practice, in more recent years there have been several emerging examples of applications in commercial products including 3D-printed footwear, parametric pattern and texture design, and the use of componentry optimized around individual user anatomy similar to the use case we will focus on in this paper. In the field of architecture, however, it is more established [1] with numerous high-profile architects employing generative design methods such as Thomas Heatherwick and the late Zaha Hadid. This can be attributed to a number of factors including the ease of parameterization in building design and the longer lead times associated with projects [2]. Inputting an appropriate set of design parameters to an algorithm with the ability to create variations in form allows a greater range of options to be explored in a much shorter time than could be achieved manually. As the functional requirements of products, particularly those that are tailored for a highly specialized goal, can present huge complexity challenges, increased computational resource expenditure is inevitable. Subsequently, there is a significant gap in the literature that homes in on combining human-factors or human-centered design (HCD) methods with computational design methods. The terms "human-factors" and "human-centred design" have some overlap, the working distinction for this paper is that the human-factors information (e.g., ergonomic, or psychological factors) forms the basis by which a human-centred design approach can be formulated.

This work follows a case-study research approach that develops thinking in generative design approaches by exploring both past examples presented within the scholarly research work and a practical example from the PRIME-VR2 research project focused on interactive therapy devices. Thus, in this paper, we firstly review the relevant literature and describe the ways in which generative design can be incorporated into the product design development process, before outlining an emerging approach for the incorporation of generative design in specific human-centered design settings where ergonomics is a critical consideration. This approach is focused on customization, which will be a critical aspect of future product design which makes use of 3D-printing technology. Three-dimensional printing or additive manufacturing is a dynamically expanding domain of manufacturing that offers new possibilities in terms of bespoke form generation and embedded functional properties. This aligns closely with HCD thinking, as customizing designs around identified human-factors concerns becomes more viable at the manufacturing stages facilitated by additive manufacturing processes [3]. Though HCD has broader implications such as improved usability, functionality, or sustainability, customization around ergonomic requirements is a distinct sub-section of the philosophy and is a vital component in any understanding of design interaction [4]. As additive technologies facilitate the creation of highly complex shapes free from some of the constraints present in other more traditional processes, there has developed a strong relationship between additive production and generative design processes [5].

With respect to this relationship, a methodology for the development of forms with highly attuned ergonomic factors will be advanced through this paper with the implicit assumption that these could be manufactured utilizing additive manufacturing methods. By considering many aspects of generative design technology, we introduce a highly relevant case study that explores the relationship between human factors and generative design methods through the creation of a bespoke product.

*Generative Design, Optimization and Generative Algorithms*

Generative design is a design process that utilizes computational methods to achieve discrete design outcomes that are explored iteratively [5]. In a generative design process, the initial constraints are established by the designer, but the form-finding algorithms applied within a CAD environment engage with these constraints to establish design options within the constraints. The designer is still required to apply insight, knowledge, and intuition in the development of the emerging design; however, this may take a different form from a typical design process. It is still necessary for a basic representation of the form to be developed based on the initial design requirements, and the rules that govern the algorithm also require critical reflection and refinement. These tasks require the creativity, intuition, and judgement associated with innovative design working [6].

An instructive means of understanding generative design is through the processes of CAD. Though the cultural and philosophical foundation of generative design was arguably laid down well before digital drawing and visualization software (biomimicry [7]), it was not until computer systems allowed for the real-time exploration of parameterized geometry and the development of algorithms that generative methods could be realized in full. Although terminology is somewhat fluid across the area of CAD, David Stasiuk's [8] breakdown of terminology is useful, differentiating "parametric", "computational", and "generative" modelling. As subsets of algorithmic modelling, parametric modelling is the most basic and is used in standard solid modelling CAD software. Computational and generative modelling approaches are more complex, allowing the system to accommodate and adapt with added intelligence such as material constraints or simulation data. While there is significant overlap between the two, "computational design" is a broader description encompassing any form of design that may be assisted by computation-based form-finding methods. "Generative design" is a narrower description but can be thought of as a subset of computational design where the solutions are more open-ended and will emerge from the solving process. The most advanced form-finding generative algorithms will not only use multiple forms of data and intelligence to inform the design, but they will also iterate different options and converge on a solution that has been optimized across a number of predefined constraints.

A range of tools have been developed to integrate computational and generative design technologies into user-friendly interfaces. Many commercially available CAD packages that utilize traditional parametric modelling methods now have integrated tools that may aid or work in conjunction with generative strategies. Creo Parametric, SolidWorks, and Fusion360 all contain a variety of optimization tools based on finite element analytics that can allow designers to generate forms optimized around particular inputs. While the scope of these tools varies from software to software, generally these programs contain the requisite building blocks for generative design workflows, such as the Pareto plug-in for SolidWorks or the new preconfigured tools present in Fusion360 which offer laymen the chance to explore some of the basic outputs of generative methods. Real command over generative processes however comes from more intricate control over the software coding and visual programming languages which have been developed to address this very need.

Visual programming languages are a distinct set of tools that allow users to connect "nodes" with pre-defined functionalities, meaning that the user does not need an intimate knowledge of programming languages such as Python or C#. Such tools have allowed for new ways of generating CAD models by providing greater control of the overall geometric construction; thereby, the process of design can be more dynamic as opposed to a series of static operations. Mistakes can be easily located within the hierarchy of nodes, and real-time explorations of form can be carried out [9]. Widely used visual programming tools include Dynamo, MEL, MaxScript, and Blueprints, all of which use the standard node and wire format. Grasshopper is one of the most widely used tools for design work and runs within the Rhino 3D CAD environment. Grasshopper can be differentiated from other generative design modules that are available by its use of visual scripting, where form-generating algorithms can be freely explored and constructed with direct feedback to the CAD visualization engines within Rhino. Notably, it offers a range of plug-ins that support generative design and optimization methods. CAD models created in this environment are truly parametric, and if constructed appropriately offer an extremely high degree of control and significant power in the generation of complex free-form geometry that is not easily achievable through conventional CAD modelling. The Grasshopper definition is built with the option of multiple variations which are implicitly possible within the design space; these can then be explored by editing specific values which relate to the parametric build, using high level design definitions, data, image processing, or number sliders within the interface. Any changes to a definition can be visualized in real-time, allowing for the rapid exploration of the possible design space. Once an algorithm has been constructed to generate geometry, this geometry can then be "baked" into the Rhino CAD environment, in which it can be edited independent of the generative algorithm. The use of Grasshopper played a central role in the case study that will be presented in the sections to follow.

While the generative processes will differ on a case-by-case basis with different design problems requiring different approaches, this does not hamper us in mapping a general process. Though generative design has evolved to be a distinct tool, its foundational work is bound up with mathematical optimization theory—work which was fused with CAD technology in the 1980s—and it is here that the process should first be understood.

The influential work on optimization by Bendsøe and Kikuchi [10] describes the basic problem: how best to lay out material within a given domain or space envelope. Indeed, they described three categories of structural optimization: (a) sizing optimization, where the basic design is engineered in terms of the connectivity of its elements; (b) shape optimization, where the optimization is engineered around a fixed topology—in the 2D case this means that a set of 2D areas or domains has a fixed topology but changeable boundaries within the plane; and (c) topological optimization, where only the domain loads and displacement constraints are specified and material may be freely assigned to each location in the design domain (2D or 3D, see Figure 1). These methods get more advanced in succession with a topological optimization generated from an understanding of design functions, as Bendsøe and Kikuchi (p. 2) describe: "The topology, shape, and size of the structure are not represented by standard parametric functions but by a set of distributed functions defined on a fixed design domain".

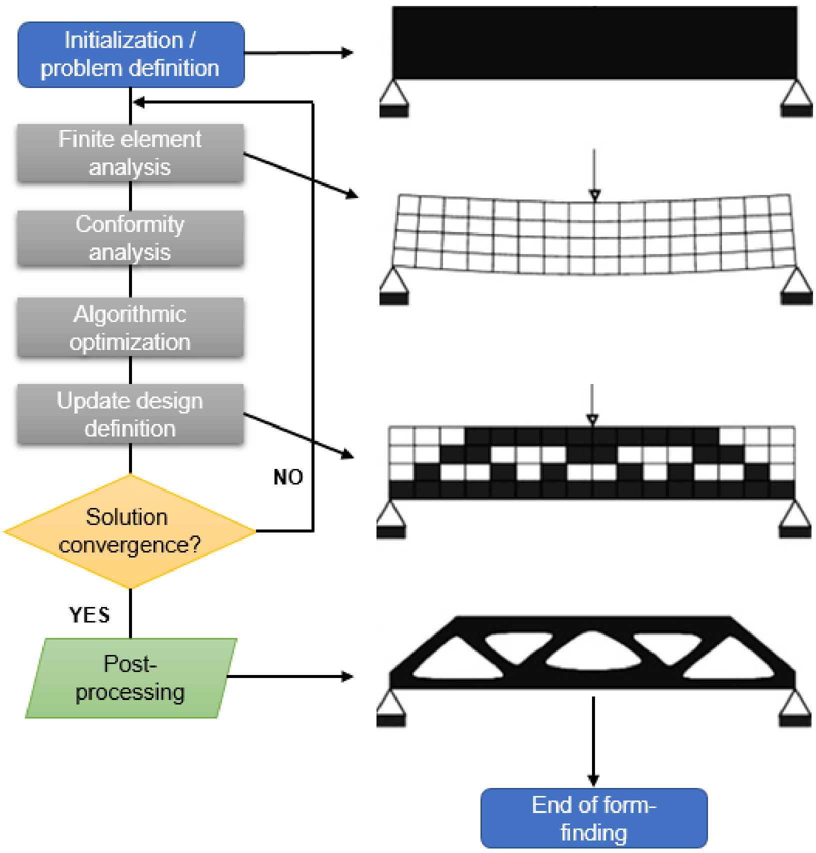


**Figure 1.** Process of form finding through topology optimization.

The topological optimization of materials is a mathematical method that can change the material layout in order to satisfy particular design goals—loading constraints and/or displacement goals for example. Using this method, structural compliances can be established within an envelope of geometry, combined with information regarding desired displacement goals which may relate to a desired function, and an "optimized" structure can be generated, often in conjunction with finite-element-methods (FEM) which are used to test for compliance or stiffness at each step of optimization iteration. Critically, this links to generative design development approaches because the foundational logic is similar, whereby a design domain is defined, and a material form structure adapts to particular inputs. In the example above, the input is the loading force on the bridge structure and the form converges on a viable solution. In an interesting study, Barbieri and Muzzupappa [11] directly compare the distinctions between topology optimization methods and generative methods. The link to the digitization of design work is made plain, with both systems reliant on CAD, CAM, and numerical control software. Crucially, topology optimization is described as a "methodology", and by contrast, generative tools are described as "artificial-intelligence-based algorithms". As demonstrated in a case study comparing solution outputs for the design of an automotive component, topology optimization methods can produce definitive answers to input conditions, i.e., what is objectively the best structure for these conditions? Generative methods have a more open solution space, where multiple options may be generated that still conform to the specific solution domain. The core benefit of these methods is the saving of material whilst achieving a specified structural strength in the design. For this reason, these methods are widely used within particular domains such as architectural engineering, automotive design, and aerospace [12]. Following Sigmund and Maute's [13] useful summary of the different optimization methods is instructive as there is a significant overlap with generative strategies, as pointed out by Mountstephens and Teo [14]. These subsequent approaches are described in Table 1.

Each method approaches the optimization in a different way. Density and evolutionary approaches use nodal-based design variables, differing from level set methods which are often used in conjunction with topological derivative methods. Hybrid methods are becoming more widely applied, such as the combination of level set and shape derivatives [15]. Although a comprehensive overview of these various methods is not the scope of this paper, a detailed summary of them can be found in the large study from Bendsøe and Sigmund [16] and from Sigmund and Maute as mentioned earlier.

**Table 1.** Summary of topological optimization methods.

| | Method | Description |
|---|---|---|
| **(1)** | **Density based methods** | The process involves splitting a structure down into microscale voids and optimizing material distribution (density) based on given design constraints. The method is highly developed and notable variants include the Simplified Isotropic Material with Penalization (SIMP) and Rational Approximation of Material Properties (RAMP) approaches [16,17] |
| **(2)** | **Evolutionary approaches** | Finite element analysis is utilized to train optimization software in an evolutionary fashion to follow particular material distribution paths [18] |
| **(3)** | **Topological derivative methods** | Applications of functional shape derivatives with respect to microscale changes in shape topology, such as adding small defects, e.g., seeding points or infinitesimal holes [19] |
| **(4)** | **Level set methods** | The structure under optimization is implicitly represented by a moving boundary embedded in a scalar function (known as the "level set" function) of a higher dimensionality. The method is flexible in handling complex topological change [20] |
| **(5)** | **Phase field methods** | The method developed as a way to represent the surface dynamics of phase-transition phenomena such as solid-liquid transitions. By utilizing the approach, perimeter control can be implemented which enables optimization [21] |

In essence, generative design is an off shoot from the initial optimization methods that are now highly developed, though many of the theoretical underpinnings still apply. We can summarize the process step-by-step.

1. **Conceptualization and ideation:** Initial design work that establishes the product or component to be created. The extent of design detailing can vary at this stage but often only a loose spatial envelope and the key functionalities are defined.

2. **Design domain definition:** The generative design algorithm must operate within a particular design "space" in order to advance particular solutions. Defining the design domain requires the designer to decide upon the boundaries of the design. Often this includes defining areas of material stiffness or flexibility, establishing a spatial profile in which the forms should be generated, or defining a functional profile.
3. **Method selection:** Building algorithms requires command over the optimization and topological transformation methods. Defining these facilitates the algorithmic design work.
4. **Integration of design domain definition with concepts using CAD environment:** Once the constraints have been adequately set up, the design domain definitions established in stage two can be advanced and integrated into a CAD environment allowing the generation of models.
5. **Generation of CAD models:** Within the selected CAD environment, models will be generated that conform to the defined design constraints. Most CAD environment will allow secondary editing after the generative stages.

## 2. Generative Design in the Design Process

Generative approaches are generally associated with conceptualization and early form development; Elgin [22], for instance, has looked at how generative design can be applied to product development and highlighted the need for "quantitative methodologies" similar to those used in Kasei engineering strategies (see [23]). Similar work [24] describes where and how generative design can fit into the conceptualization phase; the work provides a detailed description of the steps involved in executing generative design in CAD. Brossard et al. [25], in an overview of the field, suggest how algorithms can be applied throughout the design process. This includes improved performance vs. cost in detailed design, assessing geometries for manufacturability (particularly in relation to additive manufacturing), supporting optimization, and in enhancing procurement and tendering processes.

Some recent research has explored workflows in which generative design can be applied within a conventional industrial design context. Other work [26] describes different ways in which generative design can be applied for product design, with specific examples including Voronoi patterning for product bodies, procedural networks for geometric formations, and the use of Fusion 360's Generative Design module for the creation of a ceiling lamp. Similar work has explored the problems with systematizing such processes by looking at two distinct application cases: an engineering design case (a metal disc dispenser driven by technological constraints) and an industrial design case (a camera holder driven by user requirements) [27].

Reverse engineering can also be applied within this framework through the analysis of existing parametric models. For example, an existing CAD part can be analyzed and reconfigured with respect to no parameters and new information. Fantini and others [28] explored the use of reverse engineering, generative design, and additive manufacturing in the realization of a customized foot orthosis, for instance. Work from other sources has also detailed how generative design can be applied within the context of early-stage product development. GenoForm for instance, a plug-in for SolidWorks, is used by Krish [24] to demonstrate variations in a basic design and conceptually and practically separate Generative Design Methods (GDM) and Genetic Algorithm Design (GADES), suggesting the first as a much simpler method. After an algorithm has been defined (constraints set within an overall architecture), it can be modified to give design variations. This is a process of semi-automated form finding where the constraints that the designer has inputted articulate with the generative algorithms.

A key issue in the development of generative design methods is a kind of "barrier to entry" and a limited adoption within industrial design settings. A range of issues have been identified when considering how best to implement generative design methods into these "traditional" settings. These include unfamiliarity with the technology or an unwillingness to accept the benefits and process changes that the technology may bring within a specific design context. Additionally, the technology requires a steep learning curve and removes a degree of autonomy and volition from the designer or design team.

We can consider several examples whereby generative methods were shown to influence design outcomes. Brossard et al. [25] showed that in several design task situations, generative design has proven to be more efficient than teams of humans. This is also evident in work by Krish [24] where he implies that generative design will provide a method of quickly transferring conceptual designs to the further development phase—here, generative design acts as a useful next step in a development cycle. This is clearly a function of the nature of the design task, but generative design methods can notably benefit efficiency, although they may have deleterious effects for the employment of designers, for instance. Brossard et al. add that a sufficient level of design thinking will still be required to select the final design, with others highlighting that current CAD systems do not support the exploration of ambiguous concepts which are crucial in early design phases.

### *2.1. Generative Design, HCD and Ergonomics*

It is useful at this juncture to explore human-centered design, thinking in more detail about how this may advance strategies in generative design development. Human-centered design is a practical design approach that has its roots in considerations around ergonomics and safety standards [29] and is now increasingly applied in industrial and service design strategies. Giacomin [4], for example, defines HCD as comprising a number of approaches including user-centered evaluations and understanding the experience of use. Safety paradigms are also of critical concern, with many researchers exploring how human-factors knowledge can contribute to improved safety standards in the context of manufacturing and design theories (see [30]). Other work has linked a human-factors understanding with an ontology of human technology interactions that encompass human-machine, human-technology, human-system, and technology-system relationships [31]. Offshoots of human-factors theory have included design for emotion and design semantics. Don Norman [32], for example, has extensively explored the relationship between artefacts and psychological experiences. Desmet and Heckert [33] have also offered several models to understand relations between humans, objects, and emotions, with a recent theory expanding this to include concepts such as "wellbeing" [34]. Other theorists including Klaus Krippendorff [35] and Nathan Crilly [36] have studied how artefacts convey meanings to users and a physiological understanding of information processing is critical to well-developed products, particularly those involving digital interfaces.

The question for us is how HCD thinking can interact and articulate with generative design development tasks. Though generative design is often focused on the creation of static and solid objects, it can also be used to create more dynamic objects that interact directly with human anatomy in a direct engagement with human-factors information. While this is still an emerging area of research, there are a number of interesting examples to which we can refer. Prominent industrial designer and researcher Neri Oxman [37], for example, has developed a set of tools utilizing "digital morphogenesis" methods. These methods utilize a biological approach to form building, integrating biological processes of growth into the algorithmic generation tools. Her product outcomes use ergonomic intelligence to deliver structures that conform around discrete pressure points, providing improved user comfort. In an interesting paper by Crescenzio et al. [38], generative design is used to explore the issues of surface hygiene highlighted by the COVID-19 pandemic. A range of hand-free door handles are developed utilizing a Grasshopper workflow that generates an integrated off-shoot structure facilitating the hands-free functional expansion.

Other prominent areas of research include the production of prosthetics, whereby generative methods have been employed to explore more interesting aesthetics that allow a user to feel more comfortable while retaining the full functionality of the prosthetic. In this sense, GD is being utilized for another core human-factors dimension, that of aesthetic experience, product identity, and design emotion (see [39]). Some researchers have used generative design methods in order to develop more reliable prosthetics designs for amputees (see [40]). This is still an experimental area of medicine, and it is not yet widespread, but research work in this area has expanded in the last decade.

Additive manufacturing is frequently cited as a key driver in much of this field (see [5]). As the constraints of other manufacturing methods are removed, the exploration of the design space can be broadened significantly. Wang and others [41] review the status of additive technologies within the creation of customized orthotics and prosthetics and the use of computational methods to derive the product forms. From this, they derive a framework for the use of additive technologies which is instructive in its relation to generative methods for the exploration of ergonomically highly attuned medical devices. By adapting this process model (Figure 2), we can see how ergonomic data inputs can integrate with multiple strands of design intelligence including macro and micro level interrogation which will be explored further in the next section. Crucially, additive processes provide a freedom in which these design inputs can come together to present viable solutions as the constrains upon what is possible to build are removed.

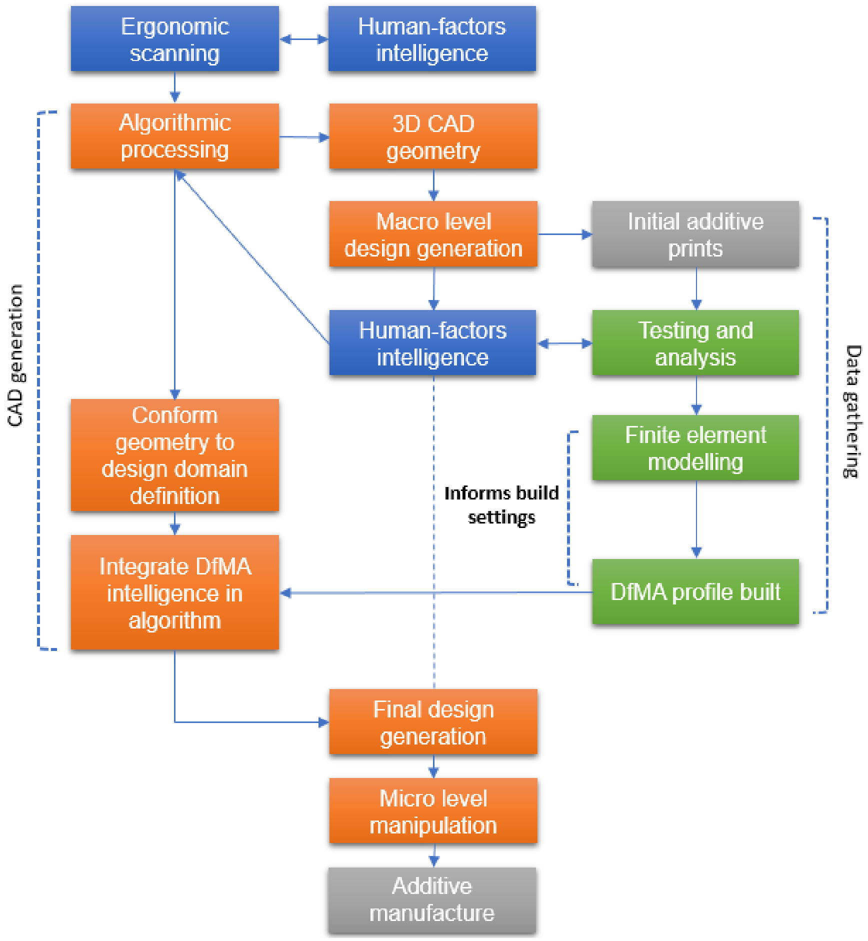


**Figure 2.** Strategy for AM production for designs derived from anatomical scans, adapted from Wang et al. [41].

A strategy similar to this is employed by Li and Tanaka (2018) [42], who use Rhino-Grasshopper programming to produce bespoke orthotics designed to immobilize fractures in hands. The paper illustrates a "proof-of-concept" for using a Voronoi optimization approach for the creation of bespoke structures attuned to user anatomy. Similar work has been produced by Abby Paterson, who has extensively explored the generation of unique splints for injury recovery [43]. Her approach uses optimizations similar to Voronoi but has created a bespoke CAD tool facilitating control over particular aspects of the build—

material thickness, for instance. This is a textbook example of the work emerging in the field of bespoke medical treatment.

### 2.2. Summary of Literature

We can see that the different research efforts discussed have utilized a wide array of design tools with many relying on more traditionally understood optimization methods as opposed to truly generative methods. The key factor missing in these discussions is how this could be translated into a workflow for generative design. While the overarching goals of the various projects are different, they are all interested in the interface between form and human ergonomics, with the caveat that advanced additive manufacturing methods facilitate the production of complex and ornate parts.

The literature indicates that this is an evolving area. When and how in the design process algorithms are implemented, how this affects the creative workflow, and what the exact outcomes are in terms of form and functionality are still very much open questions. Some examples have taken a "purer" approach with much stronger emphasis on algorithmic control, and others have used hybrid approaches that combine traditional design methods and algorithmic methods. The software tools and mechanics of algorithm construction are key concerns with regards to generating clear design outputs. The literature shows that the principal software environments currently used are Fusion360 and Grasshopper due to the flexibility of the controls offered to the designer and the relatively user-friendly interfaces that facilitate the algorithmic definitions. In addition, a number of bespoke tools that either supplement existing tools or represent completely new tools have been created by several research efforts. Furthermore, generative design overlaps significantly with work in design optimization, notably methods of topological optimization. While optimization work is similar, we can draw a conceptual boundary between the two as generative design is in essence much more open ended and adaptive.

In general terms, there is a lack of research literature relating to connecting generative design efforts with human-factors considerations that constitutes a significant gap in conceptual thinking that could benefit designers interested in human-factors. Though some researchers are producing significant innovations in this area, it still lacks systematic study. Crucially, a lack of material was identified in terms of configuring generative algorithms around key human-factors ergonomic requirements. This is a distinct problem given how the human-centered aspects of products are now an increasingly vital concern for designers. While some research efforts have developed tools for designing around human ergonomics, there is less focus on the fundamental steps to get there, i.e., a methodology for a human-factors-led generative design. The next sections will explore this in further detail by firstly looking at how generative design interacts with the traditionally understood design processes, laying the foundation for the development of an updated generative design methodology.

## 3. A Generative Design Workflow for Attuned Ergonomics

Following the research proposal set out in the introduction, we can now begin to develop a model in which generative design can be situated for further examination as part of the case study. This research method facilitates a broad exploration of the problem and then, through the use of a detailed case-study, we can establish the practical implications of these novel contributions in thinking.

While the benefits in terms of design space exploration and efficiency are well understood, there is less exploration of what specific relationship-enhanced HCD thinking within generative methods could bring to the design process—it may even prove decisive in the success or failure of a bespoke product design. Our model introduced in Figure 3 and the case study presented with Section 4 show one approach for embedding discrete ergonomic information into a generative design strategy. The approach relies on understanding the structural, behavioral, and functional relationships of the product being generated. These definitions have proved critical in our case study as the complexity of the product in question necessitates a conceptual understanding that includes not only formal properties of object geometry but also areas where articulation with proposed functions and flexibility are vital. Therefore, a feasible strategy for HCD-based generative design must always approach the problem through the lens of the abstract classes of structure, behavior, and functions. This allows a systematic build of the algorithms used in the design generation whereby a logical hierarchy has been established, delivering designs that are not only geometrically viable but also attuned to the behavioral schema of the product interactions.

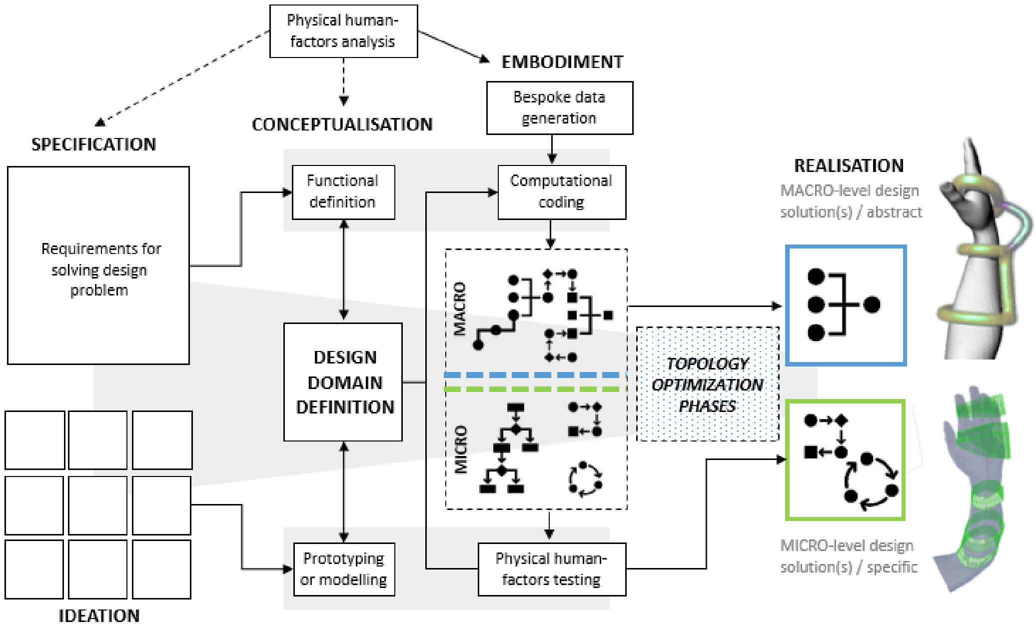


**Figure 3.** Generative design methodology incorporating macro and micro algorithmic stages.

While it is ergonomics that has been focused on in this example (delivering a feasible architecture for a functional therapy-simulating controller), it is reasonable to expand the scope of this framework to include other aspects of HCD thinking. Properties such as color and sound may also be interesting to incorporate into generative design frameworks. Work from Mothersill and Bove [44] has, for instance, incorporated information from studies in emotional perception to create a topology of emotive forms that has been directly utilized for generative design work.

When developing a methodology around generative design, it is important to consider how such a process may differ from those more traditionally understood design methodologies and philosophies. The reader should also note that this methodology utilizes the PRIME-VR2 research project as its point of reference but is intended to be a generalizable model that may apply to many generative design efforts. Many well-known and widely used methodologies such as the British Design Council's "double diamond" or Ulrich and Eppinger's [45] have a variety of benefits and drawbacks. The double diamond for instance is high-level and generic, providing the design team lots of freedoms within the approach. By contrast, Pugh's methodology is much more prescriptive and focuses the design work by applying sets of constraints around various elements such as materials and costs. These methodologies have provided guidance to designers working within the traditionally understood paradigms of artefact creation. Philosophically, these paradigms are ontologically hylomorphic (see [46]); they assume that the primary role of creation or construction comes from within a designer-agent. In this sense, the process of product creation is linear, whereby form is generated with direct interactions with known constraints and boundaries.

Generative design expressly challenges this paradigm by removing the dimension of direct designer intentionality. As we have explored in the previous sections, generative methods use the power of algorithms to explore the "design space". While this space is intentionally configured through the definition of constraints, many aspects of designer agency are removed in favor of computational methods that systematically interrogate the possible geometric formations that fit the predefined constraints. Though some processes of generative design are more removed from ideation processes, others use ideation as a means of foregrounding and structuring the generative phases. This is the approach we have chosen to adopt here in this work. Wearable interactive devices are ideal as a case study, as their design involves a close analysis of human anatomy with respect to generating customized product forms. The descriptions of the methodology will remain high-level with the intention that future researchers or practitioners may use this work as a guide for their own objectives during generative design.

1. **Specification/ideation stage:** The core requirements for the product are set out within the specifications which run in conjunction to initial ideation. This is perhaps the most well-aligned element to the traditional design methodologies, whereby sets of specifications and design concepts would be explored iteratively.
2. **Conceptualization stage:** The pre-algorithm phase, in which the functionality of the concepts are explored in the abstract. This stage may also include sketch modelling or other forms of prototyping that help to examine which design features need the most computational resources.
3. **Embodiment stage:** The algorithmic stage, in which the abstract understandings of form and functionality explored in the first two stages become refined and are built into an algorithmic definition. The algorithmic definition has a macro stage detailing the broad form features, and a micro stage detailing the functionality and design detailing.
4. **Realization stage:** The finalization of design after generative design explorations. The generative solutions will be focused on the macro (structural) level or the micro (detailing) level.

This reframing of the design process to incorporate emerging trends and understandings of generative approaches is novel. Firstly, we can see how the initial stages of the process remain more-or-less unchanged in terms of establishing a design specification. In our model, we have presented a loose relationship between a human-factors analysis and the specification definition, as there will usually be a consideration of high-level ergonomic requirements when a design project is initiated. This loose relationship is also present for the conceptualization phase where the broad parameters for the product form are set out, and the design domain definition is initiated including the establishment of abstract classes in order to define the relationship between the elements using graphing protocols.

Bespoke human-factors data have greater influence in the process where the computational coding is utilized to generate unique solutions. PRIME-VR2 has utilized mesh data from anatomical scans of the user′s arms and hands, and this is integrated directly into the pre-defined architecture.

The design solutions are finally realized within CAD visual programming tools which lend themselves particularly well to generative design workflows. Our model has highlighted macro level and micro level solutions which are differentiated by how far the solution is attuned to the human-factors inputs. A macro solution, for example, may define a broad set of spatial boundaries, and a micro level solution will incorporate more design intelligence with a more sophisticated topology optimization phase.

## 4. Application to Product Development Process

### *4.1. Micro and Macro Level Design*

With a view to creating a human-centered approach that can deal with complex information such as the anatomical requirements for wearable devices, we can demarcate the design problem in order to manage this flexibility. Programmatically, from a required functional performance through to the granular detail of the geometry of components, we can split the algorithmic approach into two broad categories: macro-level and micro-level. Each level deals with a distinct set of design problems. The macro-level deals with the processing of an overall architecture. Within the context of a human-centered approach, this would entail processing the user's biomechanical data into a target system architecture including a spatial embedding of this connected structure around the user's target anatomy. It is the macro-level algorithm that relates these abstract definitions to the spatial embedding. This spatial embedding involves arranging the components and their network of connectivity around the geometry and biomechanics of the individual product user. The macro-level graph representation supplies all the necessary boundary conditions to the sub-problem tackled by the micro-level algorithm, along with its control volume—the definition of the space within which the element under consideration can exist.

This subsequently allows us to pass from a high-level design problem at the macro level to a range of micro-level algorithms. We can then derive a complete description of the sub-problem of generating the form for a part of the enclosure which must, e.g., allow bending around a particular axis under a certain force or, say, be entirely rigid under normal operation. With respect to the description, we can relate the macro-micro bilateral to the problem of human-centered design thinking more concretely. The macro-level design relates to the breakdown of the broad ergonomics, i.e., the geometry that specifically relates to the human bodies likely to interact with the product. This could range from a very specific group to something intended for broad use within a population.

Specification and initial ideation take place at the beginning of the generative design process and establish the rough boundaries of the design. These rough boundaries allow for the elimination of a vast number of possible but not-suitable configurations, while establishing a basis in which to explore possible and suitable configurations. The specifications are set through the conventional means of research, and the potential solutions are explored in relation to these specifications with the assumption that these solutions are not optimal until all of the human-factors and design ontology intelligence has been inputted into the algorithm.

### *4.2. Conceptulisation Stage: Graphing and SBF Approach*

For computational approaches, algorithms are necessary to govern the overall topology of the structure in relation to key ergonomic data, but a series of additional rules were created for the generation of detailed structure based on target functionality. Within the context of generative design, functional definition is a key factor for a successful algorithmic design outcome [16]. This allows the algorithm to work to achieve the desired goal within the defined boundaries and creates the necessary constraints. The functional definition utilized for generative design algorithms has a specific ontological status which influences how they are implemented programmatically. Critically, this relates to an understanding of abstract classes in a Structure-Behavior-Function (SBF) design ontology [47] that allow us to understand the spatial boundaries and the functional behaviors that might exist within those boundaries. The behavior aspect is composed of a series of states of the product system with transitions between them. Each transition will have a function which implements it with any input and output variables. Each function may then in turn be composed of a series of behaviors. In this way, functions may be decomposed with ever-increasing granularity; but how does this help with form generation and the algorithmic approach? Tools such as GraphML which facilitate a machine-readable representation of product structure are a useful starting point. These tools make explicit the connectivity network of the sub-assemblies and components within a system. This is one method in which the system architecture can be described in terms of the relationships between the form, behaviors, and functions of a given product, providing a useful architecture for establishing a human-centered framework for form generation.

Referring to related functions allows checks to be carried out on the viability of the current design implementation against its intended overall function. Having this kind of machine-readable representation of the design intent and implementation allows it to be parsed and interrogated in different ways. By developing a clear framework that tracks from the highest-level goals and functions of the product system down to the component and material level of the product's implementation and fabrication, we obtain a workflow that is robust to change.

In applying this to the design problem of bespoke wearable controllers, we can see how the boundaries of the system can be grouped into abstract classes that have specific behaviors and functions. Figure 4 shows this in more detail, where a hypothetical controller design can be parsed into distinct sections or classes. The volume bounded between the inner and outer surfaces is the "thickness volume" of which there is a corresponding inner surface and contained volume and outer surface. These difference abstractions of the hypothetical form elements can be used to establish the placement of componentry and the interactions between sub-assemblies. By creating class definitions which have member variables including 3D surfaces and volumes, we are able to then move from an abstract representation of the design intent to one which can exist in a 3D CAD environment.

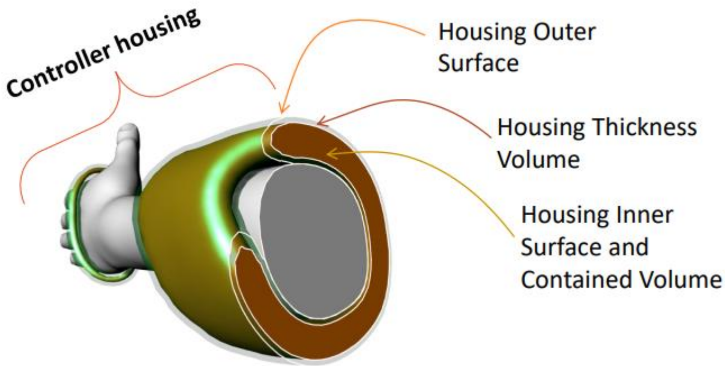


**Figure 4.** Diagrammatic relationship between the abstract volumes and surfaces of the controller housing class.

### 4.3. Embodiment Stage: Macro–Level SBF Solutions

With the intention of making the controller design highly attuned to ergonomics, our workflow incorporates the use of 3D scanners in order to acquire data on user anatomy. This is a direct input that differentiates this methodology as human-centered, drawing directly from discrete ergonomic properties to inform the design. Where this example utilizes ergonomic data, other forms of inputs could be used in replacement or as a complement.

Making use of 3D scanning and derived knowledge about the location of the user's joints, we can then map the abstract SBF graph onto a spatial arrangement around the user's arm. Figure 5 shows some hypothetical arrangements informed by the abstract class structure previously shown in Figure 4. Once in place, we can use this spatially embedded graph to partition the space around the user's arm and allocate the space available to each element or sub-assembly. Crucially, the degrees of freedom, spatial bounds, and mechanical performance required of any deformable, compliant elements will be specified at this stage. Supplementary information such as the user's weight, age, and gender can be used to complement the ergonomic information and help to fill in any unknown information. For the purposes of this research, a standard arm mesh available for Rhino was utilized in order to demonstrate the approach.

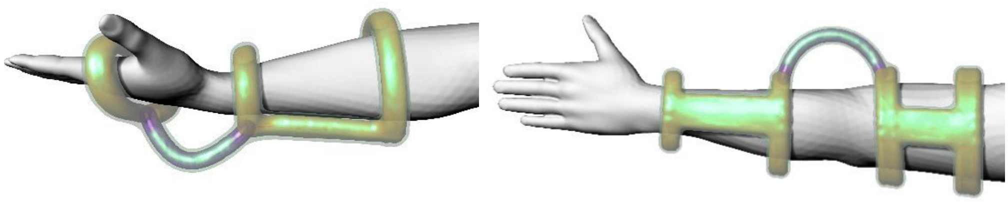

**Figure 5.** Form generations developed from abstract class definitions and ergonomic inputs.

We can then anchor the abstract graph structure to its relevant points on the user's arm utilizing scan mesh data. The initial results are shown in Figure 6 (left), where each node in the graph is given a bounding sphere of the same radius. In this way, a physical structure to the product system that is valid in its adjacency and connectivity of functional components can emerge from this form-finding process, regardless of the particular anthropometry of the individual intended user. Whilst this spatial embedding of the structure may still seem abstract, the connectivity of the elements to the bone structure of the arm and its manifold mesh surface allows for the generation of CAD geometry through a variety of means by manifesting multiple points of reference.

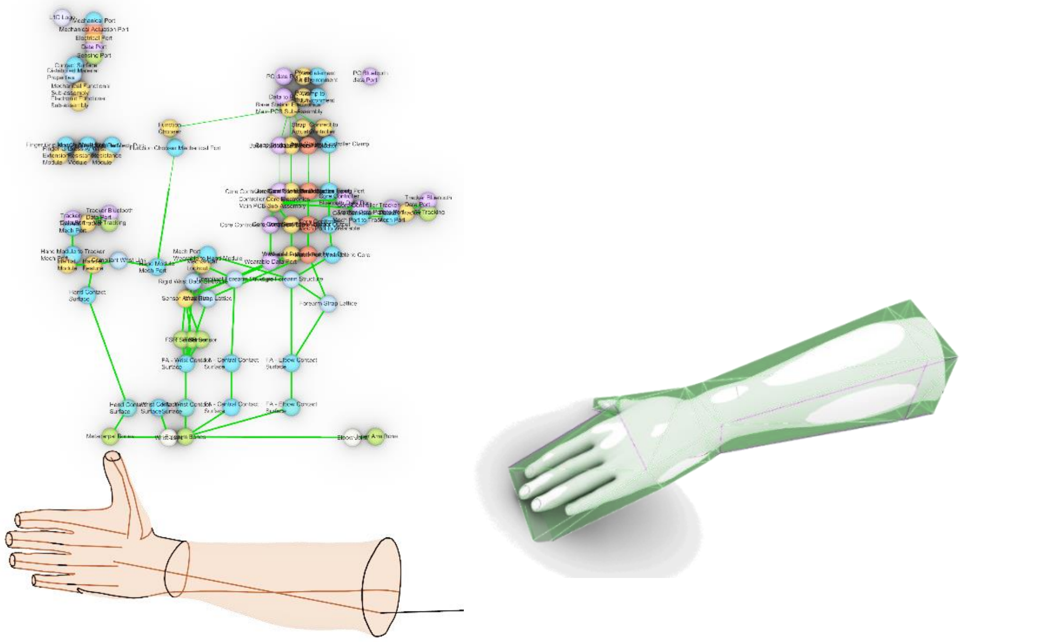

**Figure 6.** (**Left**) The structure graph network. (**Right**) Network loaded directly into the CAD environment adjacent to the arm mesh. The structure graph is arranged around the arm mesh.

The next step here is to take the scan data as an input and integrate it with the data structure representing the joints and orientation of the user's bones. By filtering the named nodes in the abstract graph representation, the number and connectivity of features which contact the user's limb can be extracted and used to drive the component placement step. Knowing the location for the custom elements that will interact directly with the user, their relationship to the user's anthropometry, the biomechanics, and the context for these elements within the product assembly, CAD models can be generated with respect to this human-factors data. The interactive physics engine "Kangaroo" inside Grasshopper was utilized to undertake a series of form-finding processes to arrive at a layout geometry for the bespoke features, taking the form of straps that can wrap around a specific user's forearm and wrist, presented as a series of surfaces with unique dimensions (Figure 7). Another script is used to "unroll" the surfaces from around the scan mesh. The output of this stage of the algorithmic approach is a series of straps composed of planar facets. The face angle between each facet and its neighbors will also be unique with respect to the ergonomic profile.

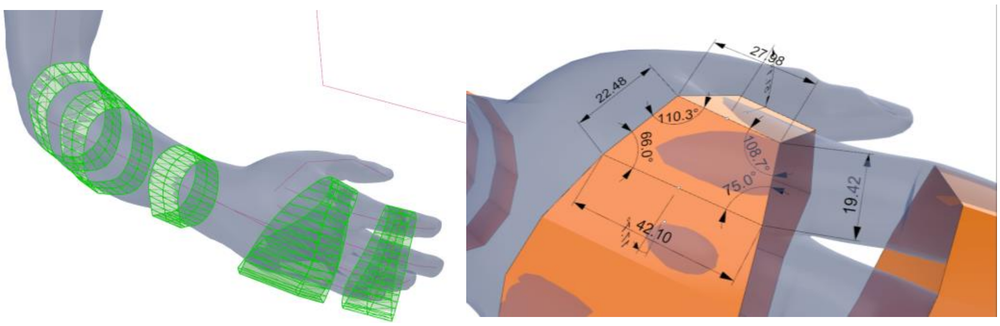


**Figure 7.** Unique dimensions of generated strap geometry.

### 4.4. Realization Stage: Generation of Device Concepts

In essence, the graphing of the relationships between function and structure facilitates the identification of critical points of interaction around the anthropometric data. From here, the crucial sub-problems that support the creation of local mechanical features, attuned around the ergonomic constraints, can then be defined. The mechanism pictured in Figure 8 has been created as a fully parametric model in Grasshopper for PRIME-VR2 and is a clear "proof-of-concept" for the human-factors based design methodology we are exploring here. The image on the left shows how the spatial relationships of the device have been informed by a user's scan mesh, with the image on the right showing an elaborated design which includes componentry and more complex mechanical elements worked into the design after the spatial definitions have been set.

**Figure 8.** An initial arrangement of a linkage resulting in a rotation around a virtual axis aligned to the user's wrist.

### 4.5. Implicaions for Design Practice

This case study shows how generative design development can be attuned around a specific human-centered workflow integrating ergonomic data and domain-specific intelligence regarding the functional goals of the object in question. This raises question of how these methods can influence wider design practice and whether a wide adoption is feasible in the current climate.

Part of what has been discussed herein is the significant conceptual shift in design thinking required to implement generative design development work effectively. In the case study, we presented the problem of design domains and the relations between elements. A system such as this is not widely used within traditionally understood design methodologies such as Pugh's [48] that essentially assume that the design solution is arrived at by a slow process of problem solving. Generative design requires a different philosophy, whereby design solutions are optimized around a defined set of constraints and critically remove direct command over the solution space. The design process is, as a result, entirely reconfigured and incorporates the modern tools of CAD, CAM, and digital optimization strategies as part of the initial conditions for successful design work.

Immediately, this has implications for the delivery of design teaching—should the pedagogic models be expanded to include CAD visual programming workflows such as

Grasshopper, for example? Some researchers have argued for this very thing (see [9,49]), as it may enhance the possibilities open to new designers and push the boundaries of innovation. Certainly, as this work has argued, HCD can benefit hugely from generative methods at both the macro-structural and the micro-detailed design levels. As other researchers such as Abby Patterson [43] have demonstrated, generative tools have huge applications in bespoke products for medicine, and combined with good interface design and software control, the tools could be widely used and broadly intelligible.

This in turn raises questions of adoption feasibility—how do such systems integrate with the current design practices that are already entrenched within industry today? As we have noted, often the problem is "cultural", where particular design approaches are viewed as superior simply because of their widespread use. A case study such as our PRIME-VR2 controller, though, may be a convincing exemplar in how generative methods could enhance innovation in HCD and perhaps broaden conversations in how industrial designers approach complex human-factors problems.

## 5. Conclusions

This paper has set out a workflow and methodology for creating a human-centered generative design algorithm utilizing a case study from PRIME-VR2. PRIME-VR2 is seeking to reinvent physical therapy through the use of bespoke virtual reality gaming controllers that replicate therapy motions. The complex and demanding nature of the design tasks meant that this was an ideal candidate for the exploration of applying human-centered design thinking in a generative design context.

After exploring some of the theoretical and practical underpinning of generative design approaches including topological optimization and visual programming languages, the PRIME-VR2 case study in which a work-through of the algorithmic build for the generation of a bespoke controller design is elucidated. Critically, ergonomic factors are used as a key input for the generation of the forms. Using data gathered from anatomical scans, ergonomic information is systematically integrated into a form-building algorithm that has been methodically structured around the logic of SBF ontology. The algorithm was built within the Rhino-Grasshopper visual programming language. Furthermore, the rigorous structuring of the algorithm facilitated the creation of coherent designs that were not only geometrically viable but also functional. In theory, any given anatomical scan, provided the data was complete enough, could be inputted into the algorithm to generate a form highly attuned to specific human factors.

This case study provides useful insights into how discrete human-factors information can be fused with a generative design algorithm to create a framework and workflow for HCD-based generative design. As human factors become a more important area of design and the use of discrete data and design intelligence becomes more common within generative design use, it is critical that clear methodologies for understanding generative methods at an abstract level are presented. In turn, this provides space in which to explore these workflows at large and develop stronger and more intuitive software tools in which designers can fully realize the potential of generative methods. While the processes of generative design (and particularly those integrating human-factors data) still require skill and technical knowledge, the demonstration of the principles through the presented case study and methodological model provides a tantalizing vision for generative design approaches that are expressly structured around human needs.

**Author Contributions:** L.U.: Primary writing, structuring, and formatting; A.W.: secondary writing and primary review; B.L.: generative design case-study, visuals, and secondary writing; C.F.: secondary writing. All authors have read and agreed to the published version of the manuscript.

**Funding:** This project has been funded by the European Commission as part of the H2020 program, under the grant agreement 856998.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References


1. Terzidis, K. *Algorithmic Architecture*, 1st ed.; Routledge: London, UK, 2006.
2. Caetano, I.; Santos, L.; Leitão, A.M. Computational design in architecture: Defining parametric, generative, and algorithmic design. *Front. Archit. Res.* **2020**, *9*, 287–300. [CrossRef]
3. Celaschi, F. Advanced design-driven approaches for an Industry 4.0 framework: The human-centred dimension of the digital industrial revolution. *Strateg. Des. Res. J.* **2017**, *10*, 97–104. [CrossRef]
4. Giacomin, J. What Is Human Centred Design? *Des. J.* **2014**, *17*, 606–623. [CrossRef]
5. Wu, J.; Quian, X.; Wang, M.Y. Advances in generative design. *Comput. Aided Des.* **2019**, *116*, 102733. [CrossRef]
6. Aghaei Meibodi, M. Generative Design Exploration: Computation and Material Practice. Ph.D. Thesis, KTH Royal Institute of Technology, Stockholm, Sweden, 2016.
7. Benyus, J.M. *Biomimicry: Innovation Inspired by Nature*; Harper Collins: London, UK, 1997.
8. Stasiuk, D. Design Modeling Terminology. 2018. Available online: https://archinate.files.wordpress.com/2018/06/dstasiuk-design-modeling-terminology1.pdf (accessed on 2 March 2022).
9. Celani, G.; Vaz, C.E. CAD Scripting and Visual Programming Languages for Implementing Computational Design Concepts: A Comparison from a Pedagogical Point of View. *Int. J. Archit. Comput.* **2012**, *10*, 121–137. [CrossRef]
10. Bendsøe, M.P.; Kikuchi, N. Generating optimal topologies in structural design using a homogenization method. *Appl. Mech. Eng.* **1988**, *71*, 197–224. [CrossRef]
11. Barbieri, L.; Muzzupappa, M. Performance-Driven Engineering Design Approaches Based on Generative Design and Topology Optimization Tools: A Comparative Study. *Appl. Sci.* **2022**, *12*, 2106. [CrossRef]
12. Yildiz, A.R.; Kaya, N.; Ozturk, F.; Alankus, O. Optimal design of vehicle components using topology design and optimisation. *Int. J. Veh. Des.* **2004**, *34*, 387–398. [CrossRef]
13. Sigmund, O.; Maute, K. Topology optimization approaches. *Struct. Multidiscip. Optim.* **2013**, *48*, 1031–1055. [CrossRef]
14. Mountstephens, J.; Teo, J. Progress and Challenges in Generative Product Design: A Review of Systems. *Computers* **2020**, *9*, 80. [CrossRef]
15. Yamada, T.; Izui, K.; Nishiwaki, S.; Takezawa, A. A topology optimization method based on the level set method incorporating a fictitious interface energy. *Comput. Methods Appl. Mech. Eng.* **2010**, *199*, 45–48. [CrossRef]
16. Bendsøe, M.P.; Sigmund, O. Material interpolation schemes in topology optimization. *Arch. Appl. Mech.* **1999**, *69*, 635–654. [CrossRef]
17. Bendsøe, M.P. Optimal shape design as a material distribution problem. *Struct. Optim.* **1989**, *1*, 193–202. [CrossRef]
18. Mattheck, C.; Burkhardt, S. A new method of structural shape optimization based on biological growth. *Int. J. Fatigue* **1990**, *12*, 185–190. [CrossRef]
19. Sokolowski, J.; Zochowski, A. On Topological Derivative in Shape Optimization. Ph.D. Thesis, Technopole de Nancy-Brabois, Lorraine, France, 1997.
20. Wang, M.Y.; Wang, X.; Guo, D. A level set method for structural topology optimization. *Comput. Methods Appl. Mech. Eng.* **2003**, *192*, 227–246. [CrossRef]
21. Bourdin, B.; Chambolle, A. Design-dependent loads in topology optimization. *ESAIM Control. Optim. Calc. Var.* **2003**, *9*, 19–48. [CrossRef]
22. Elgin, M.C. The Application of Generative Design in Product Development. Ph.D. Thesis, Arizona State University, Phoenix, AZ, USA, 2019.
23. Lévy, P.P. Beyond Kansei Engineering: The Emancipation of Kansei Design. *Int. J. Des.* **2013**, *7*, 83–94.
24. Krish, S. A practical generative design method. *CAD Comput. Aided Des.* **2011**, *43*, 88–100. [CrossRef]
25. Brossard, M.; Gatto, G.; Gentile, A.; Merle, T.; Wlezien, C. How Generative Design Could Reshape the Future of Product Development. Mckinsey Co. 2020. Available online: https://www.mckinsey.com/business-functions/operations/our-insights/how-generative-design-could-reshape-the-future-of-product-development (accessed on 25 March 2022).
26. Lobos, A. Applying Generative Systems to Product Design. In Proceedings of the XXII Generative Art Conference—GA2019, Rome, Italy, 19–21 December 2019.
27. Nordin, A. Challenges in the industrial implementation of generative design systems: An exploratory study. *Artif. Intell. Eng. Des. Anal. Manuf.* **2018**, *32*, 16–31. [CrossRef]
28. Fantini, M.; De Crescenzio, F.; Brognara, L.; Baldini, N. Design and Rapid Manufacturing of a customized foot orthosis: A first methodological study. In *Advances on Mechanics, Design Engineering and Manufacturing: Lecture Notes in Mechanical Engineering*; Eynard, B., Nigrelli, V., Oliveri, S., Peris-Fajarnes, G., Rizzuti, S., Eds.; Springer: Cham, Switzerland, 2017. [CrossRef]
29. Karwowski, W. Ergonomics and human factors: The paradigms for science, engineering, design, technology and management of human-compatible systems. *Ergonomics* **2005**, *48*, 436–463. [CrossRef]

30. Fargnoli, M. Design for Safety and Human Factors in Industrial Engineering: A review towards a unified framework. In Proceedings of the 11th Annual International Conference on Industrial Engineering and Operations Management, Singapore, 7–11 March 2021.
31. Sadeghi, L.; Dantan, J.; Siadat, A.; Marsot, J. Design for human safety in manufacturing systems: Applications of design theories, methodologies, tools and techniques. *J. Eng. Des.* **2016**, *27*, 844–877. [CrossRef]
32. Norman, D.A. *Emotional Design: Why We Love (or Hate) Everyday Things*; Basic Books: New York, NY, USA, 2004.
33. Desmet, P.M.A. A multilayered model of product emotions. *Des. J.* **2003**, *6*, 4–13. [CrossRef]
34. Desmet, P.M.A.; Pohlmeyer, A.E. Positive design: An introduction to design for subjective well-being. *Int. J. Des.* **2013**, *7*, 5–19.
35. Krippendorff, K. *The Semantic Turn: A New Foundation for Design*; CRC Press: Boca Raton, FL, USA, 2005.
36. Crilly, N.; Moultrie, J.; Clarkson, P.J. Seeing things: Consumer response to the visual domain in product design. *Des. Stud.* **2004**, *25*, 547–577. [CrossRef]
37. Oxman, N. *Material-Based Design Computation*; MIT Press: Cambridge, MA, USA, 2010.
38. De Crescenzio, F.; Fantini, M.; Asllani, E. Generative design of 3D printed hands-free door handles for reduction of contagion risk in public buildings. *Int. J. Interact. Des. Manuf.* **2022**, *16*, 253–261. [CrossRef]
39. Desmet, P.; Hekkert, P. Framework of product experience. *Int. J. Des.* **2007**, *1*, 1.
40. Sansoni, S.; Wodehouse, A.; McFadyen, A.K.; Buis, A. The aesthetic appeal of prosthetic limbs and the Uncanny Valley: The role of personal characteristics in attraction. *Int. J. Des.* **2015**, *9*, 67–81.
41. Wang, Y.; Tan, Q.; Pu, F.; Boone, D.; Zhang, M. A Review of the Application of Additive Manufacturing in Prosthetic and Orthotic Clinics from a Biomechanical Perspective. *Engineering* **2020**, *6*, 1258–1266. [CrossRef]
42. Li, J.; Tanaka, H. Rapid customization system for 3D-printed splint using programmable modeling technique—A practical approach. *3D Print. Med.* **2018**, *4*, 1–21. [CrossRef]
43. Paterson, A. Digitisation of the splinting process: Exploration and evaluation of a computer aided design approach to support additive manufacture. Ph.D. Thesis, Loughborough University, Loughborough, UK, 2013. Available online: https://hdl.handle.net/2134/13021 (accessed on 25 March 2022).
44. Mothersill, P.; Bove, V.M., Jr. The EmotiveModeler: An emotive form design CAD tool. In Proceedings of the CHI 2015: Extended Abstracts Publication of the 33rd Annual CHI Conference on Human Factors in Computing Systems, Seoul, Korea, 18–23 April 2015; ACM: New York, NY, USA, 2015; pp. 339–342. [CrossRef]
45. Ulrich, K.; Eppinger, T. *Product Design and Development*; McGraw-Hill Inc.: New York, NY, USA, 1994.
46. Ainsworth, T. Form vs. matter. In *The Stanford Encyclopedia of Philosophy*; Zalta, E.N., Ed.; Metaphysics Research Lab, Center for the Study of Language and Information, Stanford University: Stanford, CA, USA, 2016. Available online: https://plato.stanford.edu/archives/spr2016/entries/form-matter (accessed on 1 March 2022).
47. Gero, J.S.; Kannengiesser, U. The Function-Behaviour-Structure ontology of design. In *An Anthology of Theories and Models of Design*; Chakrabarti, A., Blessing, L., Eds.; Springer: Berlin/Heidelberg, Germany, 2013.
48. Pugh, S. Engineering design—unscrambling the research issues. *J. Eng. Des.* **1990**, *1*, 65–72. [CrossRef]
49. Goldstein, M.H.; Sommer, J.; Buswell, N.T.; Li, X.; Sha, Z.; Demirel, H.O. Uncovering Generative Design Rationale in the Undergraduate Classroom. In Proceedings of the 2021 IEEE Frontiers in Education Conference (FIE), Lincoln, NE, USA, 13–16 October 2021.