Design Science

Affordance based interactive genetic algorithm (ABIGA)

Ivan Mata[1], Georges Fadel[1], Anthony Garland[1] and Winfried Zanker[2]

1 *Department of Mechanical Engineering, Clemson University, South Carolina, USA*
2 *Department of Mechanical Engineering, Munich University of Applied Sciences, München, Germany*

Abstract

Designers can involve users in the design process. The challenge lies in reaching multiple users and finding the best way to use their input in the design process. Affordance based design (ABD) is a design method that focuses in part on the perceived or existing interactions between the user and the artifact. The shape and physical characteristics of the product enable the user to perceive some of its affordances. The goal of this research is to use ABD, along with an optimization tool, to evolve the shape of products toward better perceived solutions using the input from users. A web application has been developed that evolves design concepts using an interactive multi-objective genetic algorithm (IGA) relying on the user assessment of product affordances. As a proof of concept, a steering wheel is designed using the application by having users rate specific affordances of solutions presented to them. The results show that the design concepts evolve toward better perceived solutions, allowing designers to explore more solutions that reflect the preferences of end users. Relationships between affordances and product design variables are also explored, revealing that specific affordances can be targeted with changes in design parameter values and highlighting the tie between physical characteristics and affordances.

Key words: affordance based design, affordance quality assessment, interactive genetic algorithm, product evolution

Received 8 September 2016
Revised 15 November 2017
Accepted 7 December 2017

Corresponding author
I. Mata
imata@g.clemson.edu

Des. Sci., vol. 4, e5
journals.cambridge.org/dsj
DOI: 10.1017/dsj.2017.30

1. Introduction

A designer should always keep the end user in mind when designing products. After all, the end user will mostly be in contact with the product. Affordance based design (ABD), introduced by Maier & Fadel (2001), describes the interactions between the end user and the artifact (artifact–user affordances, AUAs) using the concept of affordance. Although there are many product development methods, some methods are better platforms for user involvement than others. Affordance based design has the advantage of using the concept of affordance, which is used to describe the possible ways to interact with or use the product. This suggests that users can evaluate the affordances of a product by observing it. This paper explores how end-user feedback can be captured through the evaluation of perceived affordance quality and, in turn, be used to evolve the form of products toward a better satisfaction of these affordances.

1.1. Affordance based design; the affordance concept

The term affordance emerged from the field of perceptual psychology (Gibson 1979). Gibson defined it by 'The affordances of the environment are what it offers the animal, what it provides or furnishes, either for good or ill.' It was created to describe what a system (e.g., an artifact) provides to another system (e.g., a user). Norman (1988) then extended the term to aid in the design of everyday things, but he stopped short of incorporating the concept of affordance as fundamental to the design of any artifact. Maier & Fadel (2001) introduced the concept of affordance as being fundamental to engineering design and defined it as a relationship between two subsystems in which potential behaviors can occur that would not be possible with either subsystem in isolation (Maier & Fadel 2009*a*). Furthermore, these authors described the five key properties of affordances (Maier & Fadel 2009*b*): complementarity, polarity, multiplicity, quality and form dependence.

1.2. Product evolution by improving affordances

As described earlier, one of the properties of affordances is that of quality. *Quality* tells us how well a system affords a specific use or action, in this case, based on the perception of the user. This definition of quality has been embraced by other authors (Cormier, Olewnik & Lewis 2014; Ciavola, Wu & Gershenson 2015; Ben Hamida *et al.* 2016). Products often go through many iterations (e.g., vacuum cleaners, cars, power tools), and it can be assumed that their quality increases in every iteration. Gaffney, Maier & Fadel (2007) found that the evolution of these products can be explained with an improvement in the quality of their affordances.

The quality of the affordances can be assessed either by the designer of the artifact or by different types of users (e.g., manufacturing, maintenance, end users, etc.) (Maier & Fadel 2009*b*). Use of the input from end users would help the designer to know what users perceive to be a high-quality product. It should be noted that the affordance quality assessment by users is naturally subjective. A person could agree that the sitting-ability of a chair is better than that provided by a briefcase, but could still use the sitting-ability of the briefcase, not minding the inferior quality it offers. On the other hand, there could be another person who would not tolerate the sitting-ability offered by the briefcase and would rate the affordance quality of the chair significantly higher. This means that there is no correct way to assess an affordance quality as it reflects the perceptions of users.

The challenge then is to utilize that feedback from users directly in the design process. It should be noted that affordance perception is not a variable in this framework. If a user is being asked about the quality of a particular affordance, it can be assumed that the user perceives that affordance. This does not mean that users cannot perceive any other affordances besides the ones being suggested to them. With an appropriate interface, users could suggest that an affordance be added to the product.

Ciavola *et al.* (2015) found a way to integrate the concept of affordances with genetic algorithms (GAs) and make some physical characteristics of objects change according to the evaluation of the quality of their affordances. However, their research did not make use of real end users; instead, they implemented a neural network that was trained by one user and mimicked input from that user.

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

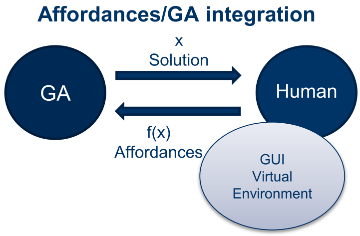

Figure 1. Affordance based design/genetic algorithm integration.

The benefit of GAs is that they allow real user assessments to be processed directly, which means that the optimization and feedback processing is done in one step. As user assessments can change over time, neural networks would need to be retrained to consider the dynamic nature of people's assessments.

This research builds on Nguyen *et al.*'s research. The ABD and GA integration (Figure 1) requires a platform where users can be reached. In the figure, 'x' represents a design solution; 'f' represents the evaluating function with an affordance quality output. To this end, a web application has been created that enables users' assessment of the quality of the affordances of a product, and their inclusion in the design process to evolve the product toward a more 'optimal solution'. Due to the fact that users interact with an optimizer (the GA), the GA becomes an interactive genetic algorithm (IGA).

The user assessment of affordance qualities is closely related to obtaining a utility function that represents the preferences from users. The use of preference models has been previously used in optimization (Orsborn & Cagan 2009; Reid, Frischknecht & Papalambros 2012), where a function that represents user preferences is determined and then is used as an optimization objective. Preference models are not needed when an IGA is implemented; use of an IGA is basically an alternative (more examples are provided later) to capturing user preferences (Ren & Papalambros 2011). The way in which user preference is usually captured is by having users select a preferred solution out of many options (Orsborn, Cagan & Boatwright 2009; Reid, Gonzalez & Papalambros 2010; Kelly *et al.* 2011; Ren & Papalambros 2011). In this paper, user preference is captured in the form of affordance quality perceptions, where the GA is in charge of evolving solutions based on the input from multiple users. The advantage of using the affordance/GA integration lies in its ability to solve multicriterion scenarios, a feature that is needed when multiple affordances can be associated with a single product.

1.3. Affordances versus requirements

Affordances should not be confused with requirements. Requirements are the objectives that the intended solution is expected to satisfy as well as the properties it must have (Pahl *et al.* 2006). There are many requirement classifications; the simplest classification has two types of requirements: demands and wishes (Pahl *et al.* 2006). Demands (in other words, constraints), which can have a fixed target value or a range of target values, must be fulfilled for the design to be acceptable. Wishes do not have to be fulfilled, but when evaluating design variants,

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

those with most wishes fulfilled, and that better meet the demands, are preferred (Pahl *et al.* 2006). Maier & Fadel (2009*a*) suggested that some requirements should be interpreted in terms of affordances, to later find solutions that offer those affordances, which in turn fulfill these specific requirements. Affordances are not requirements in the same way that functions (transformations of inputs to outputs) are not requirements. Even though functions can easily be linked to requirements (see Figure 6.6 in Pahl *et al.* (2006)), functions are considered to be the middle step between the problem definition (through abstraction of the tasks that the product has to fulfill) and the solution (structuring of the physical product through working principles). The same logic can be applied to ABD. Affordances can be a way to interpret certain requirements (abstracting requirements in terms of interactions between design entities), for which solutions have to be found in order to fulfill those requirements. An important difference from function is that affordances are perceived and therefore are related to the physical characteristics of the artifact. It should be noted also that different users may identify a multiplicity of affordances once they use the artifact. These are, most of the time, not requirements, although they may become so if the designer believes that there is an added value in highlighting that interaction.

The advantage of using affordances is that they can be easily assessed by users (due to the quality property) to determine, for example, optimal product forms. Of course, not all affordances can be assessed by users. For example, users would not be able to assess the affordances related to the rack and pinion system of a steering wheel to turn the car, simply because they cannot 'perceive' this system when using a steering wheel. However, users would be able to assess the *turn-ability* of a steering wheel, which encompasses other affordances. For example, users might be able to perceive a difference between an assisted rack and pinion system (power steering) and a non-assisted system. Even though they do not directly assess the artifact–artifact affordances (AAAs), the effects of these systems can be assessed through the turn-ability affordance (an AUA). The benefit of using the ABD framework in this scenario is the ability to synthesize the results of users' evaluations of the product through the evaluation of AUAs.

1.4. Interactive genetic algorithms

Interactive genetic algorithms are GAs where the evaluating function is substituted by human users that interact with the GA through an interface.

An IGA platform could optimize both qualitative and quantitative criteria at the same time (Brintrup *et al.* 2008). This paper discusses a platform that focuses on qualitative criteria, the affordance qualities that users perceive when examining a prototype. Furthermore, through this exercise, by studying the relationship between physical characteristics and affordance quality evolution, design knowledge acquisition that can be used in the embodiment design phase is extracted.

One of the challenges of using IGAs is the large number of evaluations that need to be made by users (Hsu & Chen 1999). As Banerjee, Quiroz & Louis (2008) pointed out, it is beneficial to have multiple users make these evaluations since they diversify the results. Researchers have worked on solutions to this problem (Hsu & Chen 1999), but these solutions involve approximating users' input for the concepts they do not assess. The affordance based interactive genetic algorithm (ABIGA) application solves this problem by allowing parallel evaluations with

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

a single CPU running the GA, specifically AMGA2 (Tiwari, Fadel & Deb 2011), a GA that works with small-size populations (∼10, as opposed to ∼100 in normal GAs).

Banerjee *et al.* (2008) implemented IGAs to evolve floorplans and widget layouts. They achieved satisfactory results within 15–20 generations/iterations. Banerjee *et al.* also found that designs evolved by a collaborative peer group were consistently rated higher on the 'originality' scale when compared with designs evolved by a single designer. This is implemented in the ABIGA framework by allowing multiple people to rate different concepts. The previously mentioned authors used a collaborative framework where individual IGA sessions could connect to each other to send and retrieve concepts. This means that such setups use multiple IGAs, each user sharing one 'best' solution per generation. The users can select solutions from other users and incorporate them in their own population.

Brintrup *et al.* (2006) and Brintrup, Ramsden & Tiwari (2007) used IGAs to optimize manufacturing plant layout designs. There were two types of IGAs: a sequential and a multi-objective IGA. The sequential IGA could be set to optimize using quantitative and qualitative objectives sequentially. The multi-objective IGA considered both qualitative and quantitative objectives simultaneously. In either case, the user was responsible for the qualitative assessments of all individuals of each generation. Their results showed faster convergence when optimizing quantitative and qualitative objectives at the same time.

Brintrup *et al.* (2008) conducted a similar experiment where they used an IGA to have users assess two subjective design parameters (comfort and liking of a chair). These can be translated to affordances of a chair. In Brintrup *et al.*'s paper, each user was responsible for evaluating all of the concepts in each experiment (all generations).

One of the limitations in previous research that implements IGAs to evolve designs is the ease of user access. In all of the research previously addressed, the computer application most likely needs to be downloaded by each of the users. This also limits the number of users that can be reached. To solve this issue, the ABIGA application was designed to be web-based. Anyone with an Internet connection and a web browser can use ABIGA. This application can be accessed using desktops, laptops, tablets and smartphones.

1.5. Description of the web application

The ABIGA is a platform that evolves product variants using the input from end users. The web application allows designers to set up design problems that can then be made available to users. The design problems require the specification of the design parameters of the artifact, the minimum and maximum values that these parameters can adopt, a virtual representation of the artifact and the list of affordances that the users will evaluate. Once a design experiment has been initialized, users can access the application through a web browser, select the experiment and evaluate the affordances of the artifact in question.

A GA, AMGA2 (Tiwari *et al.* 2011), is used to improve and evolve the design solutions available to users. The GA considers the affordances as the objectives of the problem and the design parameters as its design variables. A design concept, or solution, is defined by its set of design parameter values. These parameters are used to draw each concept for users to see and evaluate its affordances.

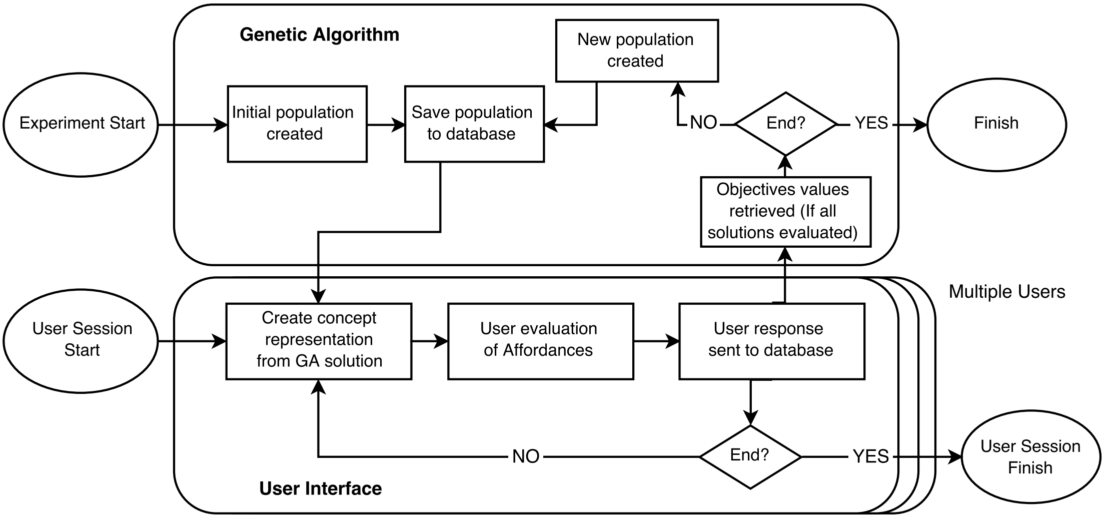

Figure 2. The ABIGA operation.

The GA is therefore solving multi-objective optimization problems. The optimization problem is defined as follows:

$$\begin{aligned}&\text{maximize}\{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x})\}\\&\text{subject to } \mathbf{x} \in S,\end{aligned}$$

where the $f_i(\mathbf{x})$ represent the affordances of the artifact evaluated by users. The variable vectors $\mathbf{x}$ belong to the non-empty feasible region $S \subset R^n$ and represent the design variable values that define a design solution.

When the experiment is started, an initial population is created by the IGA. The ABIGA creates the solutions from the IGA population, which multiple users can interact with and evaluate (see Figure 2). Once a population is evaluated in one generation, the IGA creates a new population for the next generation. Improved design concepts are created in every generation. The stopping criterion is currently set by specifying the number of function evaluations. The number of IGA generations can be determined because the number of solutions in each generation is known beforehand. The stopping criterion could also be set as an error between the population fitness average and the upper bound of that score.

1.6. The GA: AMGA2

Any multi-objective GA can be used as the optimizer in ABIGA. The archive based micro genetic algorithm 2 (AMGA2) is a multi-objective evolutionary algorithm (MOEA) that borrows concepts from multiple MOEAs (Tiwari *et al.* 2008, 2011) and was readily available to the authors. This algorithm works with a small population size and keeps an external archive of good solutions found.

The small population size (number of solutions in each GA generation) of this GA allows for a small number of users (5–10 users) to evaluate and execute the optimization. The size of the population can be as low as two times the number of objectives (Tiwari *et al.* 2011). The number of users can be large, and, as the

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

population of a generation is evaluated, a new one is generated and presented to the users. The small size of the population does mean that not many evaluations are needed to complete a GA iteration. The algorithm can have a reduced number of solutions in the population due to the use of an archive. An archive is created at the beginning of the run; once all solutions of the archive are evaluated, the GA starts to iterate with a small population size and eventually replaces better solutions in the archive.

The AMGA2 randomly regenerates the *initial population* of solutions using Latin hypercube sampling (Loh 1996) along with unbiased Knuth shuffling. A slightly modified version of differential evolution (DE) (Kukkonen & Lampinen 2005) is used as the *crossover* operator, which allows real variables to be used. In other words, DE allows the design variables to be encoded with real continuous values. The probability of *mutation* in AMGA2 is dynamic – it does not need to be tuned for specific problems; it is based on the rank of the parent solutions which can change throughout the optimization run.

1.7. Development of the web app

The ABIGA is written using Eclipse Luna (2014), an integrated development environment (IDE). Many programming languages are involved in the development of ABIGA. Java (2015) is the main language that controls most of the data. JavaServer pages are used to show data to the users, and use HTML to achieve that function. JavaScript and JQuery are used to make the web pages dynamic.

The information fed to and generated by the application is saved in a database (coded in MySQL). The Google App Engine (2015) platform is used to deploy the application. All of the services used in the application are provided by Google (database, datastore, Java runtime environment).

1.7.1. The model-view-controller architecture

The model-view-controller (MVC) architecture was followed to structure the code of the web application. This structure helps to separate the code that creates and handles the data from the code that presents the data (Hall, Brown & Chaikin 2008), taking advantage of the strengths of different technologies. For example, JavaServer pages are good at presentation, even though they allow Java code to be used; their strength is in presenting data. The strength of servlets is the processing of data, even though they allow it to be presented as well. Having the code structured with the MVC architecture allows for easy code modification and improvement of the application.

1.7.2. User interface and web pages

The user interface shows the design concept to the user as well as listing suggested affordances of the concept. Sliders with a value range from -3 to $+3$ (seven-point Likert scale) are given for each affordance so that the user can assess the quality of those affordances. It should be noted that in the current implementation, only positive affordances are being proposed to the users and the quality rating that the users will provide is explained in the instructions, as shown in Figure 3.

Other scales could have been selected. The easiest scale for users to use would be one with only three values, e.g., bad, neutral and good. However, this would

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

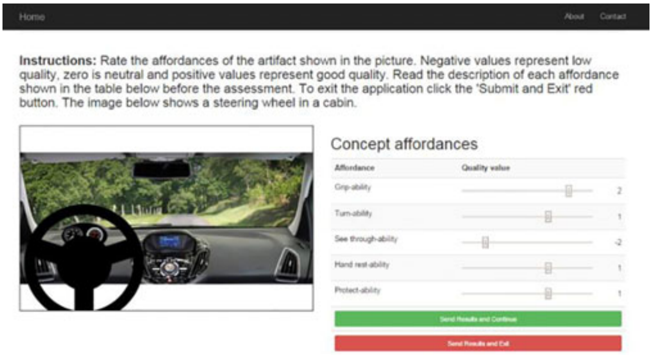

Figure 3. The ABIGA user interface.

not be sufficient to drive the GA, so a larger scale was warranted. Although a scale with a large range would be better for the GA, it would make it difficult for users to understand and differentiate quality levels. For example, in a −10 to +10 scale, users would have problems in understanding the differences between a +4 and a +6. The −3 to +3 scale was enough to drive the GA while at the same time making it easier to assess quality.

The interface of the application is based on visuals only. Some affordances would require the user to touch the artifact to effectively assess its quality. Only affordances that can be assessed through a visual interface can be implemented in the current build of the application. There are technologies that could allow users to feel the solutions through haptic feedback controllers, but this limits the number of users that can be reached. For this reason, this research focuses on visual perceptions.

Figure 3 shows a screenshot of the user interface. Instructions are provided to the user on how to evaluate the affordances of the product and what their meaning related to quality is. The center of the page is where users spend most of their time. The left section shows a 2D drawing of the product, which changes according to the solution extracted from the database. The product is placed in an environment that users are likely to see when they use it. This is to give the user some idea of the relative size of the product and its components with respect to other objects they might be familiar with. The right section has a list of the affordances of the product. Each affordance has a slider where users specify the quality of that affordance for the concept shown on the left. There are two buttons below the affordance list on the page. The green button sends the results for the current solution to the database and loads a new solution for users to evaluate. The red button sends the solution evaluations to the database and logs the user out of the experiment, which means that they do not get any more solutions to evaluate. The bottom of the page (not shown in the figure) contains the descriptions of all of the affordances of the product. Users are instructed to read these before they begin their evaluations and can always refer back to them in case they need to.

The evaluation page is not the only page that users get to interact with. There are other pages that users see before they get to the evaluation page. For example, users are required to provide some information (age, sex, education level, name initials) in a login page, they have to select the experiment they wish to be a part

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

of in an experiment selection page and, finally, they are taken to a 'thank you' page when finished.

Users are not expected to be consistent in their evaluations. User evaluation depends on the mapping between the feature parameter space (the real world, which can be described objectively) and the human psychological space. This mapping changes constantly, so even if the same feature parameters are shown to a user at different times, different evaluations may be expected. Research has been conducted that reports that this type of input noise does not affect IGA (Ohsaki, Takagi & Ohya 1998) convergence. The reason is that the IGA is not expected to converge on a point but rather an area in the solution space.

1.7.3. *How a database is used to freeze the GA*

Unlike many IGAs, ABIGA allows parallel user evaluation using a single GA running on the server. Some changes had to be made to the way in which the IGA works. Genetic algorithms, for the most part, make solution evaluations in sequence. This can become an issue for IGAs, because it would mean that if multiple users are used as evaluators, they would have to evaluate one at a time, i.e., each would have to wait for other users to finish evaluating their solutions.

This challenge was overcome in ABIGA using Google's Datastore technology. The Datastore is a schemaless NoSQL scalable storage service. The Datastore allows the storage of data objects (such as Java classes that hold data). When a new generation of solutions is created, the GA data are saved in the Datastore and the generation solutions are saved in a database. Because of this, the GA does not need to run while all of the solutions are being extracted from the database to be evaluated. This also means that multiple users can evaluate solutions at the same time, significantly improving the total user evaluation time. Once all solutions for a generation are evaluated, the data are retrieved from the Datastore and used by the IGA to generate a new set of solutions.

2. Design of a steering wheel

A steering wheel was redesigned using ABIGA. The steering wheel was modeled with five design parameters (see Figure 4) and five affordances given by the authors. It should be noted that the affordances shown in Table 2 are not the entire set of affordances needed to design the steering wheel. Other affordances are involved that cannot be evaluated by users, especially AAAs; for example, the affordances related to the steering shaft (if there is one at all) and the rack and pinion system. *The designer needs to select a set of relevant affordances that users can evaluate and that will have an impact on the form of the product*. The experiment was carried out twice. Two different groups of six users were involved in the assessment of design solutions. The initial population had 50 solutions; all subsequent generations had a population of 10 solutions each. Users were only instructed to access the web application with its URL. The stopping criterion for the IGA was the number of generations, which was set at 15 generations in both experiments based on Nguyen *et al.*'s reported results with a similar design problem (Nguyen *et al.* 2012).

The design parameters that make up the steering wheel are shown in Table 1 along with the range of values each parameter can adopt.

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

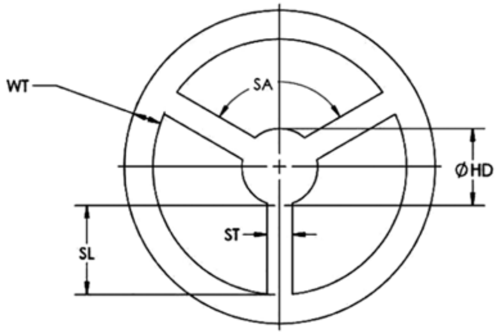

Figure 4. Steering wheel design parameters.

Table 1. Steering wheel design parameters

Design parameter	Minimum value	Maximum value
Hub diameter (HD)	30 pixels (61 mm)	70 pixels(142 mm)
Spoke length (SL)	50 pixels (102 mm)	80 pixels (163 mm)
Spoke thickness (ST)	20 pixels (41 mm)	40 pixels (82 mm)
Ring thickness (WT)	10 pixels (20.5 mm)	25 pixels (51 mm)
Top two spokes angle (SA)	50 degrees	250 degrees

Table 2. Affordance descriptions

Affordance	Description
Grip-ability	Gripping interaction between the hands of the user and the steering wheel.
Turn-ability	Interaction between the hands of the user and the steering wheel where the user rotates the steering wheel to turn the car.
SeeThrough-ability	Visual interaction between the user and the steering wheel that allows the user to see through the steering wheel. This could affect the visibility of the dashboard gauges and or the street.
HandRest-ability	Interaction between the hands of the user and the steering wheel that allows users to rest their hands on any part of the steering wheel.
Protect-ability	Interaction between the user and the steering wheel where the steering wheel protects the user in the event of a crash.

Pixels were used as the units for the design parameters (with the exception of the angle parameter) as a proof of concept; in industrial applications, designers

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

would of course use appropriate units. The value range of each parameter is chosen to match the size restrictions of the dashboard image used as the background to the steering wheel. These values can be converted to real size values when compared with the size of the dashboard used as a background. Pixels can first be converted to millimeters (1 mm $\approx$ 3.78 pixels) and then this value can be scaled back to normal size. For example, the dashboard width is 640 pixels (169.3 mm) in Figure 3. If the width of a dashboard is 1300 mm, then the scaling factor needed would be approximately 7.68 (scale factor = real size/image size).

The affordances of the steering wheel are shown in Table 2 with the descriptions shown to the users.

3. Experimental results

The ABIGA stores a lot of information for each design experiment. Every concept generated by the GA is stored in the database. This includes the design parameter values and the affordance evaluations of each solution. All of this information can be queried from the database for analysis while the experiment is running or after it is completed.

3.1. Product evolution

To check whether the solutions in the IGA improve across the different generations, the fitness of the entire population can be tracked. Since there are multiple objectives (number of affordances), the overall fitness of a solution can be reduced to the sum of all of its objectives. This is done only for easy visualization of the evolution of the product. This is not how the IGA operates. The IGA is a multi-objective algorithm that uses the rankings of each solution with respect to each objective to evolve solutions toward better ranked solutions that eventually become non-dominated, and possibly Pareto. Details on how the GA works can be found in Tiwari *et al.* (2008, 2011). The graph does show, however, the trend toward solutions that are perceived to be better by the users, considering all of the affordances. Figures 5 and 6 show the average of solution fitness values across all generations for experiments 1 and 2, respectively.

The graphs show 16 generations because the first generation is the initial population, which also has to be evaluated by users. The maximum value possible for the overall fitness is 15, as there were five affordances, each of which could have a maximum value of three. The minimum value is −15 because the lowest quality value for each affordance is −3.

A steady increase in the fitness of the population can be seen after the 12th generation in experiment 1. The results of experiment 2 show a similar behavior, starting from generation 10. This means that the solutions generated by the IGA were perceived to be better in quality than the initial population solutions as a whole. It should be noted that this does not mean that there were not any solutions rated as good solutions in earlier generations. To show how all objectives improve across generations, Figure 7 compares solutions from generation 1 with the solutions in the last generation (15). Three solutions were selected for these generations: low-, medium- and high-valued solutions.

To test whether product evolution can be attained out of chance, two experiments were performed using a random number generator (RNG) as the evaluator of quality affordance. The results are shown in Figures 8 and 9.

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

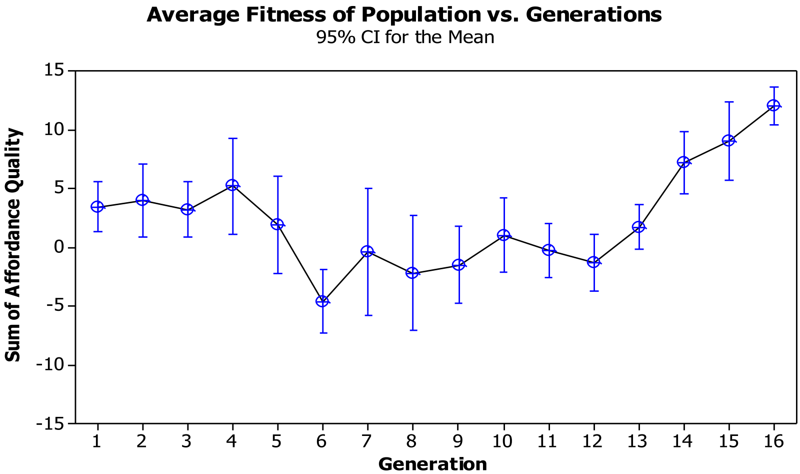

Figure 5. Steering wheel evolution; experiment 1.

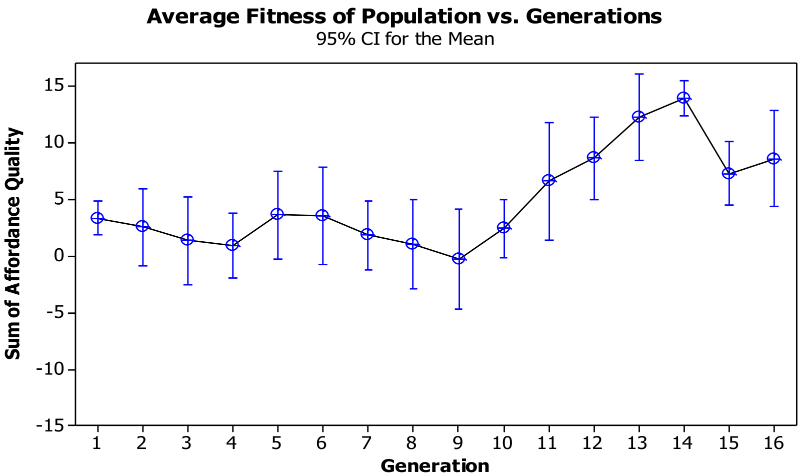

Figure 6. Steering wheel evolution; experiment 2.

Besides saving the solutions generated by the GA in the database, the best solutions are saved separately for easy access. These solutions are solutions found in the non-dominated front at the end of the optimization run. Figures 10 and 11 show some solutions generated by the IGA on the first generation and the final generation of experiment 1, respectively.

Even when ABIGA provides a set of optimized solutions, the designer still needs to perform other types of evaluations that cannot be obtained through user evaluations, most likely considering other parameters on those solutions. Instead of suggesting a specific value for each of the parameters, the designer could use value ranges obtained from the assessments of the users. To determine these ranges, the data would have to show that there are relationships between the affordances and the design parameters of the product.

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

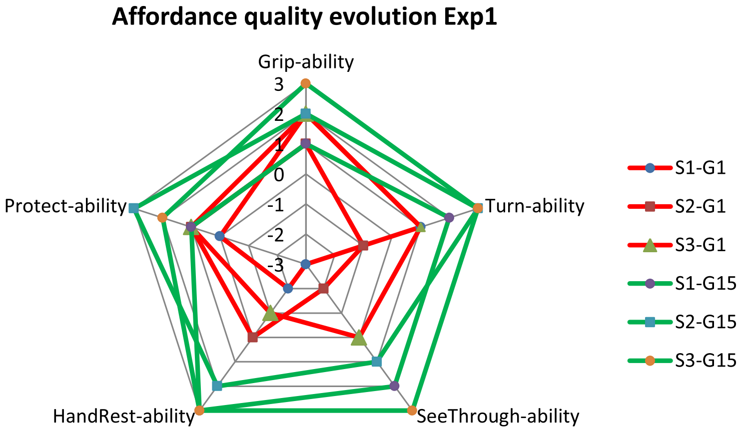

Figure 7. Affordance quality evolution; experiment 1.

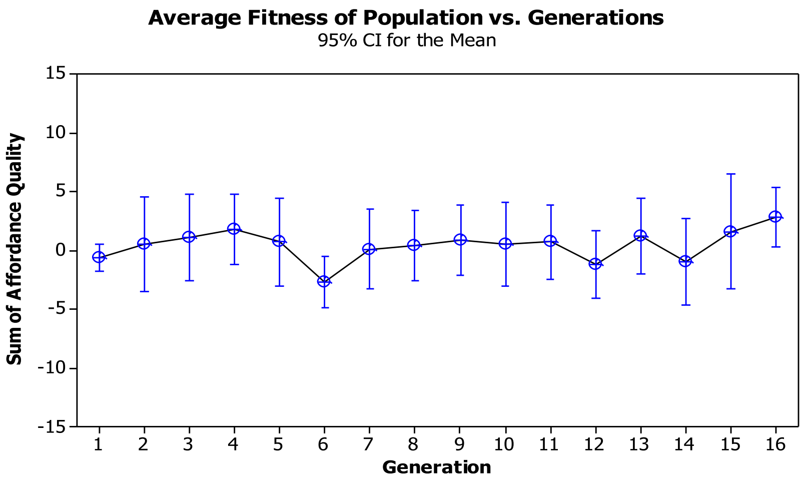

Figure 8. Steering wheel evolution RNG; input 1.

3.2. Relationships between affordance quality and design parameters

The existence of relationships between affordances and design parameters is suggested by one of the properties of affordances, *form dependency*. This property says that the affordances in a product are dependent upon the shape or geometry or physical characteristics of objects. For example, the form of a big box does not offer hand grip-ability, but if a handle is added to the sides of the box, therefore changing the geometry of the box, grip-ability is now possible due to that change. Having relationships between affordance qualities (as perceived by users) and the design parameters of the solutions would mean that the designer could select design parameter values that consistently get positive ratings by users to target specific product affordances.

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

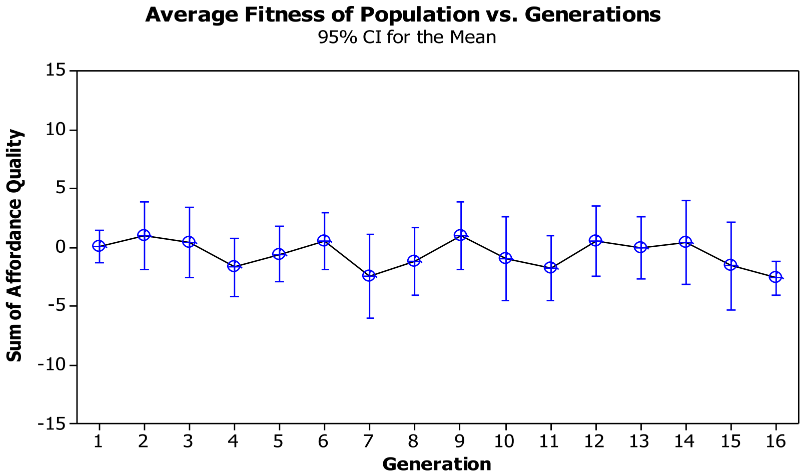

Figure 9. Steering wheel evolution RNG; input 2.

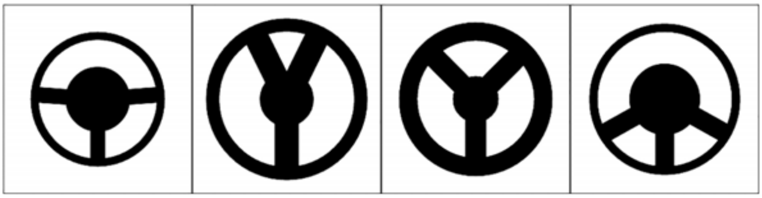

Figure 10. Subset of generation 1 solutions; experiment 1.

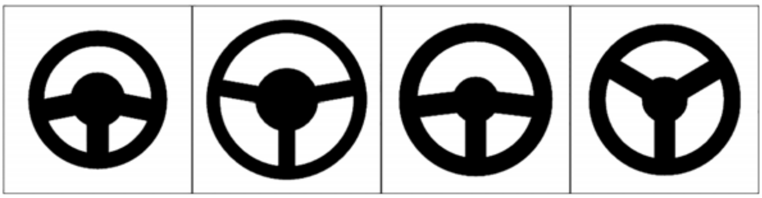

Figure 11. Subset of archive solutions; experiment 1.

The affordance quality input from users is categorical; that is, the variables are discrete, ranging from -3 to $+3$. Users were instructed that negative values meant a bad affordance quality, zero meant neutral and that positive values meant that the product had a good affordance quality. Unlike the affordance variables, the design parameter variables are continuous. Due to this discrepancy, linear regression techniques cannot be implemented between these variables to determine whether there is any relationship between them. Instead, *binary logistic regression* was implemented as it is the recommended method to test relationships between categorical and continuous data.

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

Table 3. Binary categorization of user response

Seven-point Likert scale	−3	−2	−1	0	1	2	3
Binary response		Not good (0)				Good (1)	

Table 4. Binary logistic regression p-value results; experiment 1

	HubDiameter	SpokeLength	SpokeThickness	RingThickness	TopTwoSpokesAngle
Grip-ability	0.071	0.214	0.981	0.404	0.042
Turn-ability	0.086	0.023	0.475	0.173	0.001
SeeThrough-ability	0.009	0.202	0.015	0.018	0.000
HandRest-ability	0.196	0.525	0.045	0.194	0.000
Protect-ability	0.956	0.065	0.011	0.015	0.685

The affordance variables were further categorized into a binary type of response. Table 3 summarizes how this categorization was made. The zero-valued responses were not used in the analysis.

The reason for making the affordance variables binary is because designers are interested in what users consider to be 'good' solutions. If a relationship between a design variable and an affordance exists, the binary logistic regression can tell us the design parameter values that are more likely to be perceived as positive by the users.

The logistic regression is represented by

$$\ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x, \tag{1}$$

where p is the estimated probability that a user would positively rate an affordance given a design parameter value of x. The intercept β_0 and the coefficient β_1 are given in the logistic regression results for all relationships found. With this equation and the results of the logistic regression for two variables, the designer can determine the value of the design parameter for a desired probability of acceptance.

The binary logistic regression analysis was carried out using Minitab16 (2016), and therefore not performed by ABIGA itself. All of the solutions generated by the GA are used in the binary logistic regression tests. Tables 4 and 5 show the results of design parameters tested against all affordances for both experiments (the Hosmer–Lemeshow goodness of fit test results are given in the Appendix). The p-values of the logistic regression tests are shown in the tables for each pair of design parameter and affordance. The number of observations for each binary logistic test varies according to how many data points have been omitted due to not considering the zero-valued responses. The average number of observations is 180. If the p-value is less than 0.05, then there is evidence that the design

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

Table 5. Binary logistic regression p-value results; experiment 2

	HubDiameter	SpokeLength	SpokeThickness	RingThickness	TopTwoSpokesAngle
Grip-ability	0.055	0.885	0.980	0.317	0.021
Turn-ability	0.002	0.985	0.049	0.120	0.009
SeeThrough-ability	0.001	0.000	0.030	0.002	0.000
HandRest-ability	0.012	0.152	0.638	0.653	0.000
Protect-ability	0.701	0.000	0.003	0.000	0.968

parameter contributes to the prediction of the affordance quality outcome. The significant values are highlighted in the tables.

The logistic regression coefficients, β_0 and β_1, are given in Tables 6 and 7 only for the significant relationships. These coefficients can be plugged into Eq. (1) to mathematically describe the significant relationships between affordances and design parameters. An example of how this is used is provided in the following section. It should be noted how the sign of β_1 defines whether the relationship is positive or negative. A positive sign indicates that the quality perception probability increases with an increase of the design parameter value.

4. Discussion

4.1. Product evolution

Figures 5 and 6 show how the IGA is able to gather and combine the input from users to create solutions that are perceived to be high quality by most users. However, this does not mean that the optimized solutions provided by the IGA can be directly transferred to production. As mentioned above, the designer, of course, still needs to carry out other types of analysis on these solutions to make sure that they meet other design criteria that are not represented within the affordances evaluated by the users. For example, the designer might need to perform a stress analysis on the chosen solutions to make sure that the product will not fail under usage loading. In spite of possibly obtaining design solutions that do not meet other design criteria, the information given by the IGA is valuable as it provides design variable values that are most likely to be perceived as high quality by the users.

The results of the IGA are in accordance with the findings of Nguyen *et al.* (2012), which showed that the average fitness of the population across generations reached a maximum at about 15 generations. This result is important because their steering wheel problem was single objective. The problem solved in this paper is multi-objective. This means that an increase in the number of objectives did not increase the number of generations needed to reach high population fitness averages. This suggests that an increase in the number of objectives (product affordances) may not negatively impact the number of generations that the IGA should perform to find good solutions.

The results in Figures 10 and 11 show the general trends in the design of the solutions of the initial population and the best solutions saved in the archive

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

Table 6. Experiment 1 logistic regression coefficients ($\beta_0 \mid \beta_1$)

	HubDiameter	SpokeLength	SpokeThickness	RingThickness	TopTwoSpokesAngle
Grip-ability					−0.10194 | 0.00520
Turn-ability		4.11820 | −0.04487			−0.38662 | 0.00961
SeeThrough-ability	−1.14763 | 0.03099		2.43684 | −0.06664	2.00054 | −0.08651	−2.25213 | 0.01672
HandRest-ability			2.21856 | −0.05544		−2.09646 | 0.01686
Protect-ability			−2.12545 | 0.07768	−1.55994 | 0.09727	

Table 7. Experiment 2 logistic regression coefficients ($\beta_0 \mid \beta_1$)

	HubDiameter	SpokeLength	SpokeThickness	RingThickness	TopTwoSpokesAngle
Grip-ability					−0.52910 | 0.00655
Turn-ability	4.70065 | −0.05837		−0.66798 | 0.07441		0.02348 | 0.00929
SeeThrough-ability	2.82589 | −0.04713	−4.85299 | 0.08320	2.46779 | −0.06561	2.72297 | −0.13508	−3.68685 | 0.02326
HandRest-ability	2.72101 | −0.03715				−2.67811 | 0.02058
Protect-ability		7.96893 | −0.10360	−1.96306 | 0.10407	−2.91543 | 0.25578	

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

of the IGA for experiment 1. The angle between the top two spokes has higher values in the archive than in the initial population. This makes sense as this design parameter allows users to see through the steering wheel to better appreciate the gauges on the dashboard of the car. The thickness of the ring seems to be larger in the archive than in the initial population. This suggests that users are more likely to perceive a thicker ring as being of higher quality than a thin ring, like the first solution shown in Figure 10.

4.2. The bad evaluator problem

It might seem that if a user deliberately gave solutions bad quality values, the IGA would therefore provide bad results. Due to the way in which GAs work, this is not a problem with ABIGA. To search for the best solutions, GAs combine good solutions (crossover operation) to create new solutions. This means that even if there is a user who consistently gives products bad assessments, the IGA will most likely not be too affected by those solutions. This does not mean that all of the solutions chosen for the crossover operation are good solutions; there is always a small chance that bad solutions will be chosen because the IGA will try to explore as much of the solution space as it can. The only, and unlikely, scenario that could affect the evolution of a product in ABIGA is when the entirety of users in an experiment purposefully give bad assessments.

4.3. Design parameters versus affordances correlations

As is expected in multicriteria problems, there are always tradeoffs, which means that there is no perfect solution. Because of this, there is no single solution that will be preferred by all users. Therefore, it does not make sense for the designer to expect to use specific design parameter values obtained from ABIGA in their design. After all, we would not expect that a quarter degree in the angle between the top two spokes of a steering wheel would make much of a difference to the perception of quality from users. It would make more sense if designers had ranges of values for each design parameter that they could work with so that whatever values were chosen would elicit good quality perceptions from the users. It turns out this is possible if the responses from the users show that there are relationships between the design parameters and the affordances of the product in question.

As shown earlier, using logistic regression techniques, relationships between design parameters and affordances were found. These relationships have been highlighted in Tables 4 and 5. These results suggest that designers can target specific affordances by changing the values of design parameters so that users perceive a high quality for those affordances.

It should be noted that the use of a logistic regression technique to check for these relationships inherently limits how the relationships are described. The dependent variable (affordance quality) is turned into a stochastic event (good/bad affordance quality, as described in Table 3). This is described with a density function of cumulated probabilities ranging from zero to one. This means that other models cannot be used to describe these relationships, such as quadratic or cubic models. Future experiments could be modified, making affordance quality a continuous variable to allow for other types of models to

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

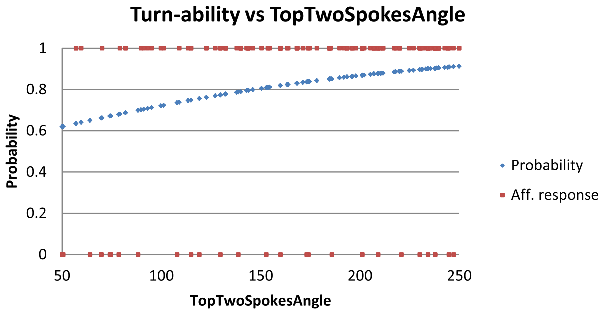

Figure 12. Turn-ability versus TopTwoSpokesAngle; experiment 2.

be fitted. Nonetheless, as will be shown later in this section, logistic regression provides valuable information to designers.

One of the relationships found was between *Turn-ability* and *TopTwo-SpokesAngle* in both experiments. By graphing the logistic regression between these variables, a lot of information can be used by the designer to improve the product. Figure 12 shows this logistic regression; the *probability* series represents the probability of a good outcome at every value of the independent variable (the design parameter). The *affordance response* series represents the categorized evaluation (user response transformed to a binary response) from users. This means that if an angle of about 150 degrees is chosen for the steering wheel, then there is about an 80% chance that a user would rate it as a good design. A good design, as mentioned earlier, represents qualities of 1 to 3 based on the seven-point Likert scale used in the experiment.

This can provide valuable information to a designer. Instead of trying to select one value for a design parameter, the designer can focus on a range of values that would make the users have a positive perception of specific affordances in the product. As mentioned earlier, the designer can use the optimized solutions obtained from experiments. However, these solutions may need to be changed to meet other design criteria. The relationships between affordances and design parameters can be used to change design parameter values and predict how users will perceive product affordances.

Besides helping designers to target specific affordances with design parameter changes, these relationships give clues on how users make use of the product. The fact that *turn-ability* is related to the angle between the top two spokes means that users use the spokes to turn the steering wheel. This information could be used to identify new affordances that could improve the usability of the steering wheel. For example, designers could improve the grip on the spokes of the steering wheel to make it easier for users to turn the wheel.

As can be seen in Table 4, design parameters can be related to multiple affordances.

Figure 13 shows the relationship between *SeeThrough-ability* and *TopTwoSpokesAngle* for experiment 1 (similar results were obtained in experiment 2). The results of this relationship make sense, as the larger this angle becomes, the easier it is to see through the steering wheel, making it easier to see the gauges on the dashboard in the cabin. *HandRest-ability* is also related to *TopTwoSpokesAngle*,

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

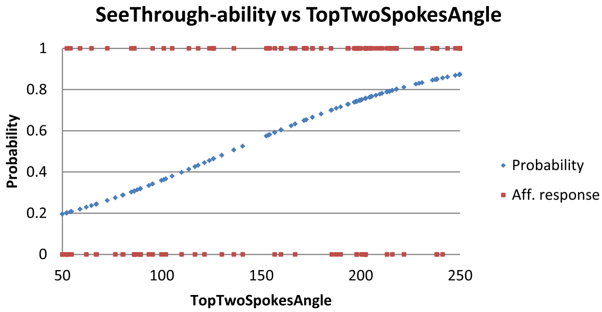

Figure 13. SeeThrough-ability/TopTwoSpokesAngle; experiment 1.

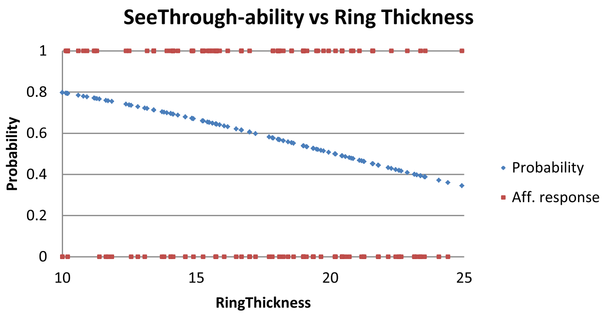

Figure 14. SeeThrough-ability/RingThickness; experiment 2.

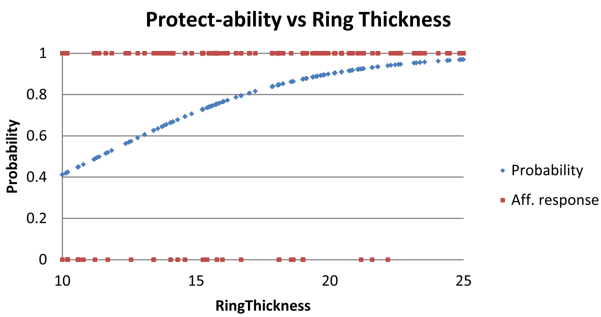

Figure 15. Protect-ability/RingThickness; experiment 2.

which means that users think of resting their hands on the spokes rather than on the ring of the steering wheel.

The thickness of the ring (*RingThickness*) is related to *SeeThrough-ability* and *Protect-ability* (see Figures 14 and 15). Unlike the relationships shown earlier, the SeeThrough-ability relationship is inversely proportional. This means that as the design parameter value increases, the probability of a good assessment decreases.

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

It should be noted that due to the fact that six users were employed in each experiment, the results do not represent the perceptions of 'all' end users, and therefore it is not expected that all experiments will yield the same relationships as seen in Tables 4 and 5. The goal of these experiments is not to characterize the perceptions of all individuals; for that to happen, larger crowds need to be used in each experiment (which is possible with ABIGA).

4.4. Design parameter values to target specific affordances

For the relationship shown in Figure 13, if the designer wants to know the angle between the top two spokes for which the probability of acceptance is 65%, Eq. (1) can be used to find it (the coefficients are given in Table 6). This gives the designer a range of values for which the probability of acceptance is equal to or higher than 65%,

$$\ln\left(\frac{0.65}{1-0.65}\right) = -2.25213 + 0.0167285x, \tag{2}$$

$$x = 171.6 \text{ degrees.} \tag{3}$$

The result shows that if the designer chooses any value between 171.6 and 250 degrees, most users would rate the *SeeThrough-ability* of the steering wheel as good. This can be carried out for all of the relationships found through logistic regression analysis, giving designers a lot of freedom when choosing values for different design parameters while making sure that their decisions will be perceived as 'good solutions' by the users. It should be noted that the equation above can only be used within the range of design parameters that was tested with users. Extrapolation cannot be performed with the logistic regression results.

4.5. Multi-objective trade-off analysis using affordance/design parameter relationships

The relationships in Figures 14 and 15 show a conflict between Protect-ability and SeeThrough-ability with respect to the same design parameter (RingThickness). The Protect-ability and RingThickness relationship is positive. This means that if the RingThickness is increased, the perceived Protect-ability quality improves, but the perceived SeeThrough-ability worsens. It is the designer's choice to favor a particular affordance. This trade-off analysis can be made using the results from these two relationships.

For example, if the probability of acceptance is set at 0.7, the RingThickness can be any value in the range (Reid *et al.* 2010; Nguyen *et al.* 2012), according to the SeeThrough-ability relationship. However, according to the Protect-ability relationship, the RingThickness value can be any value in the range (Loh 1996; Orsborn *et al.* 2009). The designer here can clearly see the trade-off in choosing a ring thickness value. In this example, the designer can choose to favor one of the conflicting affordances.

4.6. Applications of ABIGA

As suggested throughout the paper, ABIGA can be used in the design process to help designers to optimize the shape of a product considering the input from end

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

users. The tool can also be used in the redesign of products where the architecture is already defined but the design parameter values can be optimized. This paper focuses on the usability aspect of design, which is the reason why end users were selected as the type of user. However, the web application can be used on other types of users to optimize the design based on the input of that specific group. For example, manufacturing and assembling users will have their own set of affordances related to manufacturing and/or assembly. The needs of different types of users could potentially change the shape of the artifact.

4.7. Limitations of the current tool

The application optimizes the shape of the artifact. There are no changes in the topology. However, this would be possible if the designer had come up with different topologies (or configurations) and would like to know the optimal topology configuration. The features can be coded as existing (1) or not existing (0), effectively allowing the GA to work with multiple solution configurations.

The set of affordances shown to the users is currently fixed. The application can be modified to allow users to suggest affordances that might be added in the middle of the optimization run. This could expand the solution space by adding features to the products that fulfill the suggested affordances.

The tool currently does not allow for three-dimensional product representation to be shown to users. The use of three-dimensional renderings of products could improve the ability of users to assess the quality of their affordances. This will be explored in future experiments.

5. Conclusions

The results of the steering wheel example suggest that the ABD and IGA integration can be used to evolve design variants toward solutions that are perceived to be better by users. The results from ABIGA give designers a lot more information than just a set of design parameters that make up design solutions. Relationships between the affordances of the product and its design parameters can provide information on how users perceive the use of the product. This can help designers to identify more affordances that can be used to modify the solutions to improve the usability of products.

Instead of selecting one of the solutions suggested by ABIGA, designers can determine ranges of values for the different design parameters that are related to the affordances of the product. This gives designers the freedom to be able to choose multiple values for different design parameters that would make certain affordances be perceived as positive by the users. The fact that relationships between affordances and design parameters were found serves as proof that affordances indeed depend on the shapes of artifacts.

The steering wheel example is a proof of concept. The experimental results suggest this can be applied to other design problems. Moreover, different products with varying numbers of affordances and design parameters need to be tested to check the complexity of products that can be handled by ABIGA.

Appendix

Hosmer–Lemeshow goodness of fit test; experiment 1 (Chi-squared | Degrees of freedom | p-value)

	HubDiameter	SpokeLength	SpokeThickness	RingThickness	TopTwoSpokesAngle
Grip-ability	5.668 | 7 | 0.579	6.742 | 8 | 0.565	10.793 | 7 | 0.148	25.085 | 8 | 0.002	5.368 | 7 | 0.615
Turn-ability	18.316 | 7 | 0.011	13.452 | 8 | 0.097	8.517 | 7 | 0.289	4.183 | 8 | 0.840	3.994 | 7 | 0.780
SeeThrough-ability	4.169 | 7 | 0.760	11.462 | 8 | 0.177	9.508 | 7 | 0.218	2.228 | 7 | 0.946	7.654 | 7 | 0.364
HandRest-ability	11.715 | 7 | 0.110	8.019 | 8 | 0.432	8.129 | 7 | 0.321	6.342 | 8 | 0.609	15.975 | 7 | 0.025
Protect-ability	10.972 | 7 | 0.140	4.533 | 8 | 0.806	7.408 | 8 | 0.493	13.883 | 7 | 0.053	16.523 | 7 | 0.021

Hosmer–Lemeshow goodness of fit test; experiment 2 (Chi-squared | Degrees of freedom | p-value)

	HubDiameter	SpokeLength	SpokeThickness	RingThickness	TopTwoSpokesAngle
Grip-ability	4.443 | 8 | 0.815	6.454 | 8 | 0.596	5.112 | 8 | 0.746	32.41 | 7 | 0.000	6.494 | 8 | 0.592
Turn-ability	13.252 | 8 | 0.103	9.712 | 8 | 0.286	12.174 | 8 | 0.144	7.102 | 8 | 0.526	9.993 | 8 | 0.266
SeeThrough-ability	10.724 | 8 | 0.218	12.838 | 8 | 0.118	9.828 | 8 | 0.277	12.741 | 8 | 0.121	33.439 | 8 | 0.000
HandRest-ability	11.960 | 8 | 0.153	8.113 | 8 | 0.423	6.712 | 8 | 0.568	24.195 | 8 | 0.002	27.815 | 8 | 0.001
Protect-ability	22.043 | 8 | 0.005	3.730 | 8 | 0.881	31.447 | 8 | 0.000	25.039 | 7 | 0.001	8.508 | 8 | 0.385

References

Banerjee, A., **Quiroz, J. C.** & **Louis, S. J.** 2008 A model of creative design using collaborative interactive genetic algorithms. *Design Computing and Cognition '08* 397–416.

Ben Hamida, S., **Jankovic, M.**, **Curatella, F.**, **Baltay, S.**, **Callot, M.**, **Huet, A.**, **Bocquet, J.-C.** & **Sasse, S.** 2016 Value proposition design for systems and services by adapting affordance-based design. *Proceedings of ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American Society of Mechanical Engineers.*

Brintrup, A. M., **Ramsden, J.**, **Takagi, H.** & **Tiwari, A.** 2008 Ergonomic chair design by fusing qualitative and quantitative criteria using interactive genetic algorithms. *IEEE Transactions on Evolutionary Computation* **12** (3), 343–354.

Brintrup, A. M., **Ramsden, J.** & **Tiwari, A.** 2007 An interactive genetic algorithm-based framework for handling qualitative criteria in design optimization. *Computers in Industry* **58** (3), 279–291.

Brintrup, A. M., **Takagi, H.**, **Tiwari, A.** & **Ramsden, J. J.** 2006 Evaluation of sequential, multi-objective, and parallel interactive genetic algorithms for multi-objective optimization problems. *Journal of Biological Physics and Chemistry* **6** (2006), 137–146.

Ciavola, B. T., **Wu, C.** & **Gershenson, J. K.** 2015 Integrating function- and affordance-based design representations. *Journal of Mechanical Design* **137** (5), 51101.

Cormier, P., **Olewnik, A.** & **Lewis, K.** 2014 Toward a formalization of affordance modeling for engineering design. *Research in Engineering Design* **25** (3), 259–277.

Eclipse, 2014 Eclipse Luna software [Online]. Available https://projects.eclipse.org/releases/luna [Accessed: 08-Dec-2016].

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

Gaffney, E. S., **Maier, J.** & **Fadel, G.** 2007 Roles of function and affordance in the evolution of artifacts. *The Design Society Proceedings of the International Conference on Engineering Design ICED'07*, Paper no. 592.

Gibson, J. J. 1979 The theory of affordances. In *The Ecological Approach to Visual Perception*, Lawrence Erlbaum Associates, Inc.

Google, 2015 Google App Engine [Online]. Available http://code.google.com/appengine/ [Accessed: 05-Jun-2016].

Hall, M., **Brown, L.** & **Chaikin, Y.** 2008 *Core Servlets and JavaServer Pages*. Pearson Education, Inc.

Hsu, F. C. & **Chen, J.-S.** 1999 A study on multi criteria decision making model: interactive genetic algorithms approach. In *IEEE SMC'99 Conference Proceedings. 1999 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No. 99CH37028)*, pp. 634–639. IEEE.

Kelly, J. C., **Maheut, P.**, **Petiot, J.-F.** & **Papalambros, P. Y.** 2011 Incorporating user shape preference in engineering design optimisation. *Journal of Engineering Design* **22** (9), 627–650.

Kukkonen, S. & **Lampinen, J.** 2005 GDE3: the third evolution step of generalized differential evolution. *Institute of Electrical and Electronics Engineers Proceedings of the 2005 IEEE Congress on Evolutionary Computation,* vol. 1, pp. 443–450.

Loh, W. L. 1996 On Latin hypercube sampling. *Annals of Statistics* **24** (5), 2058–2080.

Maier, J. & **Fadel, G.** 2001 Affordance: the fundamental concept in engineering design. *Proceedings of ASME 2001 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American Society of Mechanical Engineers*.

Maier, J. & **Fadel, G.** 2009*a* Affordance-based design methods for innovative design, redesign and reverse engineering. *Research in Engineering Design* **20** (4), 225–239.

Maier, J. & **Fadel, G.** 2009*b* Affordance based design: a relational theory for design. *Research in Engineering Design* **20** (1), 13–27.

Minitab, 2016 Minitab [Online]. Available https://www.minitab.com/en-us/ [Accessed: 01-Jan-2016].

Nguyen, M. T., **Guarneri, P.**, **Fadel, G.** & **Mata, I.** 2012 Genetic algorithms applied to affordance based design. *Proceedings of ASME 2012 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American Society of Mechanical Engineers*.

Norman, D. A. 1988 *The Design of Everyday Things*. Currency Doubleday.

Ohsaki, M., **Takagi, H.** & **Ohya, K.** 1998 An input method using discrete fitness values for interactive GA. *Journal of Intelligent & Fuzzy Systems* **6** (1), 131–145.

Oracle, 2015 Java software [Online]. Available https://www.java.com/en/ [Accessed: 08-Dec-2016].

Orsborn, S. & **Cagan, J.** 2009 Multiagent shape grammar implementation: automatically generating form concepts according to a preference function. *Journal of Mechanical Design* **131** (12), 121007.

Orsborn, S., **Cagan, J.** & **Boatwright, P.** 2009 Quantifying aesthetic form preference in a utility function. *Journal of Mechanical Design* **131**, 61001.

Pahl, G., **Beitz, W.**, **Feldhusen, J.** & **Grote, K. H.** 2006 *Engineering Design: A Systematic Approach*. Springer.

Reid, T., **Frischknecht, B. D.** & **Papalambros, P. Y.** 2012 Perceptual attributes in product design: fuel economy and silhouette-based perceived environmental friendliness tradeoffs in automotive vehicle design. *Journal of Mechanical Design* **134** (4), 41006.

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press

Reid, T., **Gonzalez, R. D.** & **Papalambros, P. Y.** 2010 Quantification of perceived environmental friendliness for vehicle silhouette design. *Journal of Mechanical Design* **132** (10), 101010.

Ren, Y. & **Papalambros, P. Y.** 2011 A design preference elicitation query as an optimization process. *Journal of Mechanical Design* **133** (11), 111004.

Tiwari, S., **Fadel, G.** & **Deb, K.** 2011 AMGA2: improving the performance of the archive-based micro-genetic algorithm for multi-objective optimization. *Engineering Optimization* **43** (4), 37–41.

Tiwari, S., **Fadel, G.**, **Koch, P.** & **Deb, K.** 2008 AMGA: an archive-based micro genetic algorithm for multi-objective optimization. *Proceedings of the 10th Annual Genetic and Evolutionary Computation Conference (GECCO'08)*, pp. 729–736.

https://doi.org/10.1017/dsj.2017.30 Published online by Cambridge University Press