Resumen

Abstract

Dedicatoria

Dedico este trabajo a mi esposa Iliana Durn Landero por el apoyo incondicional que brindaste durante todo el proceso. Sabemos muy poco, y sin embargo es sorprendente que sepamos tanto, y es todava mas sorprendente que tan poco conocimiento nos de tanto poder. Bertrand Russell Despus de todo, cualquier tipo de conocimiento implica auto-conocimiento. Bruce Lee El conocimiento descansa no solo sobre la verdad sino tambin sobre el error. Carl Jung El conocimiento depende del tiempo, mientras que el saber no. El conocimiento es una fuente de acumulacin, de conclusin, mientras que el saber es un continuo movimiento. Bruce Lee

Agradecimientos

Agradezco al CONACYT por haberme brindado la oportunidad y el apoyo a través de la beca 256554 durante el periodo de Agosto-2011 a Agosto-2015 para concluir los estudios de Doctorado en Ciencias en Computación.

A mi director de tesis, Dr. Mario García Valdez que le tengo una enorme gratitud por haberme enseado herramientas tan valiosas para mi formación como profesional así como persona de tecnologa. Al Dr. Oscar Castillo López y tambin a la Dra. Elba Patricia Meláin por haberme brindado la oportunidad de entrar al programa de doctorado en ciencias de la computacin del Instituto Tecnolgico de Tijuana , a todos ustedes gracias por su paciencia y recomendaciones para hacer posible este proyecto de tesis.

A mi familia, por su comprensión y apoyo, en especial a mis padres, el Ing. Julian Torres Ruiz, Lic. Susana Hernández Velarde por haberme apoyado en el cuidado de mi hija Victoria Alessandra Romero Durón, ya que sin ustedes hubiera sido imposible lograr esta meta.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

It is a reality that the World Wide Web in recent years, is growing exponentially, which means the presence of millions of users on Web sites, Web applications, Web systems, etc. [1]. There is a wide variety of Web systems, where we have different users interacting with them. These users have different goals when using these Web systems. For example to serching in Google [58] for a particular topic, make a reservation for a room in a luxury resort, check their bank account or simply checking their Facebook account status [14].This variation of users represents a complex diversity as individuals []. This diversity lies in the fact that users have different skills, interests, preferences, knowlage defferent ways of thinking and learning.For this reason users most to interact with the information presented by existing Web systems.

When we intend to customize any element in Web system, it is necessary to some personal information about the user. This information is a collection of needs, characteristics, tastes, preferences among others. This information allows designers to build a representation of knowledge about the users. This is what is known as user modeling (UM).

In order to understand specific users a user model must be constructed, this can be simple as a profile where the basic knowledge recorded. Or it can be as complex as complete representation of it's characteristics, needs, interests and preferences. In order to understand specific users. The main goal of user modeling is to represent aspects of the real world of the user's in an autonomous way.

On the other hand, interactive evolutionary computation (IEC) is a branch of evolutionary computation where users become a part of the evolutionary process by replacing the fitness function; evaluating individuals of a population based on their personal preferences[13]. These evaluations are subjective according to the user point of view based on their perceptions, interests and desires.

Normally such systems require users to evaluate large amounts of individuals actively, causing them to lose interest in participating by the fatigue that is generated[13]. Nowadays some of these systems are migrating to Web technologies looking for volunteer users to collaborate in the evaluation for distribute the load and lower the fatigue. Having Web- based interactive evolutionary systems opens the possibility of linking to social networks platforms in order to involve a larger number of users to assist in the evaluation of individuals produced by these systems.

This research presents a user-cetered framework that involves several techniques such as graph-based user modeling, fuzzy logic and human-interaction in the context of a Web-based interactive evolutionary computation. The purpose of this research is to increase voluntary participation of the users using the proposed framework.

1.1 Outline

- **Chapter 2**, describes an in-depth study of current and related works, presenting a general overview of Interactive evolutionary computation and their evolution in the pass and recent years, also user model approaches and finally gamification paradigm. This study includes Interactive evolutionary computations methods and techniques to understand how they work, as well as their problems of these systems. On the other hand this study also includes user modeling methods in order to understand how to create and apply them for certain necessities. Finally we discussed a gammification usability paradigm to understand how it work and subsequently apply this for the necessities of this work.

- **Chapter 3**, describes the fundamental concepts that form the basis of the proposed user-cetered framework.

- **Chapter 4**, presents a user-centered framework for interactivity Web-based applications, this proposed framework involves different paradigms in order to increase user participation. This chapter also includes the overall explanation of how the functionality the proposed framework works.

- **Chapter 5**, the case study is presented along with the explanation of the experimentation for this work. The experiments were realized using several versions of the same application in order to see which of them have most user participation.

- **Chapter 6**, The results of the case study are presented. This chapter includes all the data obtained from the experiments, these data are presented to see which of the versions meets the best performance regarding the increase of participation of users.

- **Chapter 7**, finally the conclusions and future work of this research are exposed.

At the end, this thesis includes appendices that describe detailed technical aspects about installation EvoDrawings application on heruku The rules of the fuzzy inference system are presented in

Chapter 2

State of the art and background

This chapter describes an in-depth review of current and related works, presenting a general overview of Interactive evolutionary computation and their evolution in the pass and recent years, also user model approaches, In the same way some works were review that contain the fuzzy logic concept and finally gamification paradigm is also review.

2.1 Interactive Evolutionary Computation.

This technology is branch of evolutionary computation. Based on subjective human evaluation. Basically, this technique requires that the objective function is replaced by a person (user) [12]. Takagi defines it as "A optimization technology that uses

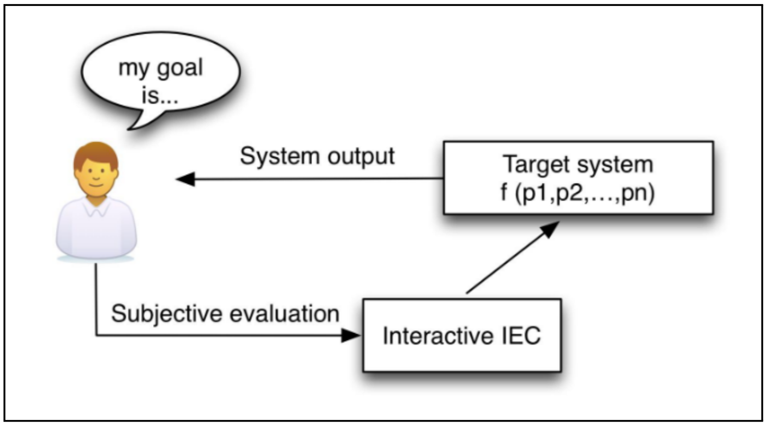

Figure 2.1: General Interactive Evolutionary Computation system based on subjective evaluation.

human evaluation in the optimization system instead of a human evaluation model. Simply stated, the interactive EC is an EC whose fitness function is replaced by a human. In figure 2.1 we show a general IEC system.

In this general IEC system we note that the user replaced the fitness function at the moment he or she interacts with the system. Logically the user needs a goal to know what is the evaluation he or she needs to perform. Finally the users receive an output and the process and start over again.

These techniques have been used in several areas of application, in particular:

- Music and sound.
- Digital Art.
- Design and editing documents

- Processing acoustic signals.
- Industrial design.
- Data mining and acquire knowledge.
- Face recognition.
- Robotics and control.

Below are some of the most outstanding works in the interactive evolutionary computing area, which are the following:

Dawkins's research was the pioneer of a significant addition to the 1990s IEC algorithms research works[16].

There is two key research approach about his field:

- **Creative Approach:** The Artificial Life (AL) was the base of creative approach. AL uses complex algorithms for biological life models emulation. To perform this task, it is needed to include some of the different techniques starting from right image treatment. Good graphic creation as well as a great music and quality sounds, [69], [71], [16], [21], [82]and [83].

- **Humanized technology approach:** The concept of humanized technology approach comes from the approach that is focused on the IEC algorithms interface, this is the research of interaction between humans and computer systems. The main goal of this was to reduce the user's fatigue and to promote the inputs and outputs of algorithms to improve the efficiency of them. IEC has made his own way in practical fields such as engineering, education,etc., [60], [84], [75], [63], [61] and [74].

Computer graphics (CG) The Biomorph of Dawkins was the first IEC research, from this research comes to many motivated works mostly about the Selfish Gene, come of these works are: [55], [45].

In Dawkins work, a conventional recursive algorithm was used as a baseline maintaining the main target of trees with an L-system (Lindenmayer). This same L-system was the base for another experiment to create 2-D CG forms insects from a system called Blind Watchmaker who used L-system angles from L-system output intuitively selected; the creation was called biomorphs. These creations reach his target with the multiple selections of the users based on their preferences; all these selections acted like a natural adaptation filter.

We can find plenty of applications and works for fractal generation and [70], [4] and [5], [43], or [2],[64][Raynal 1999] and [44], for rendering in tridimensional, [78],[11], [15][Das 1994] and [76], for generation of virtual creatures, [67], or aerodynamic surface design (wings), [53], [52] and [][NGuyen 1997].

We can discover more than one additional way to use this research in the artistic field with several applications of IEC who are used for cartoon face construction and

Figure 2.2: Panspermia.

animations matters, like Mutator [79] and [80] or [6].

The genetic programming (GP) applications offers a category called Interactive Genetic Programming (IGP) with many examples of successful application in tridimensional artwork for artistic animations or construction using mathematical equations as CAVE [15], [59] and [21], [69], [70] and [49]. As this work consequence, Panspermia or Primordial Dance was created that are presented in figure 2.2 and figure 2.3.

The artistic field is only the first step of a great IEC implementation; it is important to mention another relevant projects called Galapagos, [72], and SBART, [82]. The IEC application Galapagos Project is the exhibit in Tokio Multimedia

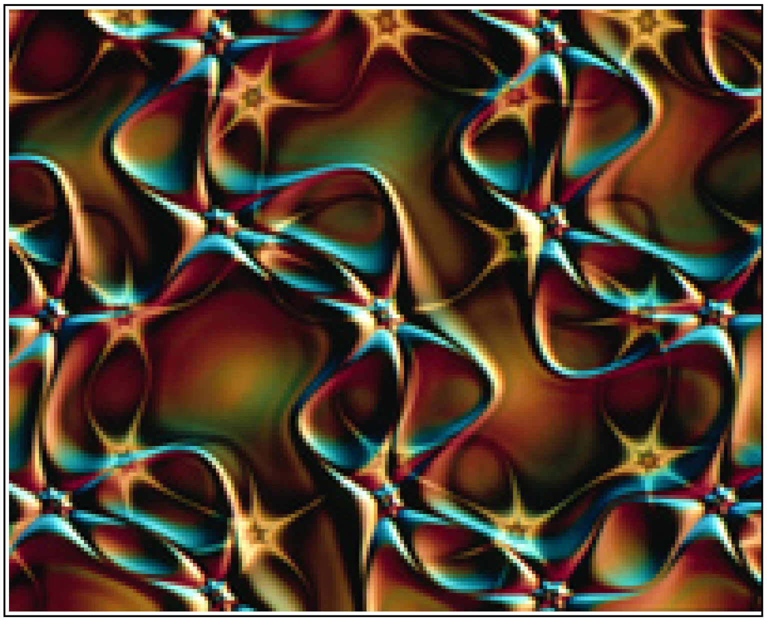

Figure 2.3: Primordial Dance.

Museum, (NTT Intercommunication Center) and this project originates engaging images to all visitors based on L-systems as we can see in figure 2.4 and figure 2.5.

There are created after one selection, to get a good solution through multiple repetitions. This action is performed with Genetic Programming (GP), after the calculation of each pixel value using trees of equations combining logarithm, maximum,

Figure 2.4: Galapagos: Tokio Multimedia Museum.

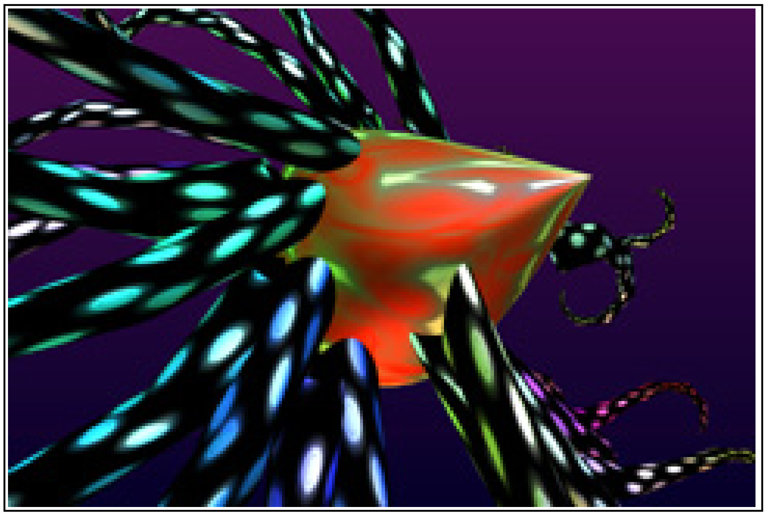

Figure 2.5: Galapagos' output sample.

and minimum, sine, root, cosine, exponential arithmetic operators. AnimationLab is found as an outstanding work who offer figures that can run or walk working with the user to receive more opportunities to be picked. A particular characteristic of all of the figures is that the figures extremities Mentioning open source works, we can find SBART as an IGP [82] tool to create graphics. SBART allow to users to evaluate 20 two-dimensional images, subsequently twenty new image has direction and angles as we can see in figure 2.6.

There are many examples for this field application as [47], [84], or [85], [41] and [42]. One of the Interactive Evolutionary Programming (IEP) artistic application was created by [2], as a fractal generation where the system allows the evolution of animations for the ones who were selected from the user, the application initially show only 10 animations to rate.

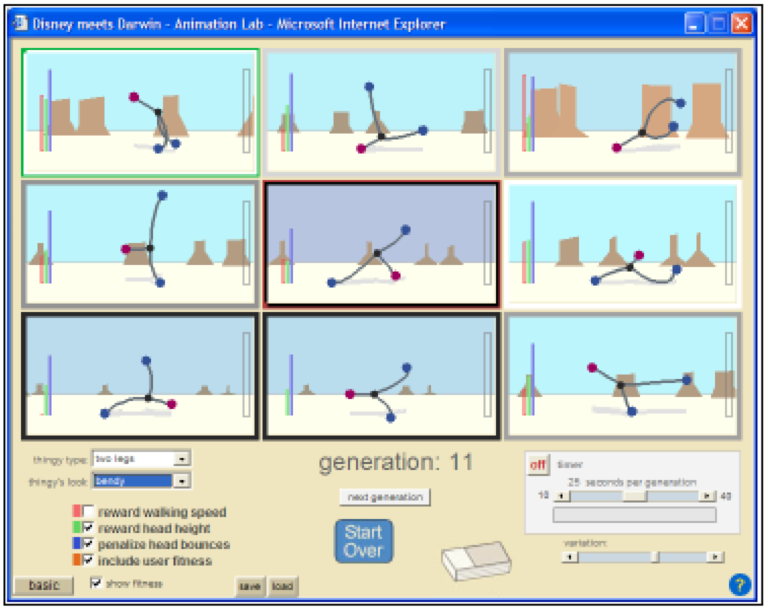

Figure 2.6: Animation Lab.

It is important to know how IEC was implemented in music generation, with several applications in this field. We will start mentioning the pioneer application GENJAM, [7], [8] or [9] and [10]. Some other attractive works are Sonomorph, [50] and [51], or SBEAT, [83], [33], [57], [81] and [24]. It is possible to hear a part of the music songs of these previously mentioned works broadcasted in the radio station WDYN. (100.1, New York, USA, WEBPage:http://www.wdyn.net/).

The IEC algorithms are the base for the functionality of the music generation systems, a visual representation of this is given in the below figure 2.7:

In 1998, a new input method for human operators of an interactive genetic algorithm to reduce the psychological weight is proposed. This method uses a discrete

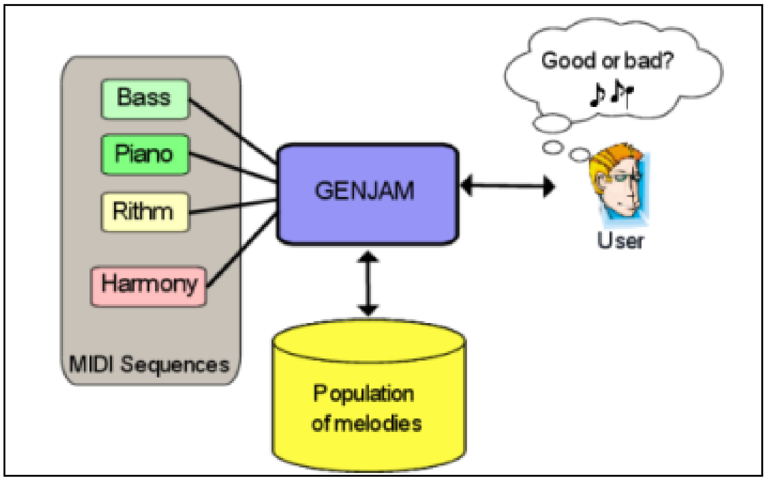

Figure 2.7: GENJAM squeme.

fitness values to reduce the psychological stress involved in the input procedure. They perform simulations to investigate the influence of the resulting quantization noise from the use of discrete values of fitness in convergence. Showing that the quantization noise does not significantly worsen in the convergence. In this method they evaluated using two subjective tests involving the task of drawing faces.

The subjective test results shows that this method significantly reduce the level of psychological stress of human interactive genetic algorithms operators [56]. Another approach, proposed to used novel method evaluation. Where the user only evaluates a satisfactory or unsatisfactory individual. These approach consider the level of sensibility of the different users to their perception of the beautiful and the ugly, and fitness is automatically calculated based on user evaluations and time. They also propose effective strategies for comparing different individuals of the same generation in uncertain fitness conditions of an individual. Where they obtains the probability of an individual dominance by use of the probability of the interval domain, and translate to a fuzzy number in a range based on -cut set [26]. They determine the dominant individual in tournament selection with size being two on base given by the probability of a particular domain. This approach was applied to an interactive evolutionary system for fashion design. In figure 1 we can see different user interfaces they used. Based on this approach, another work was de-rived. Where the approach adopt a fuzzy number described with a Gaussian membership function to express an individual's fitness. In order to compare the different individuals, they generated a fitness interval based on a cut set, and obtain the probability of interactive genetic algorithms with individual's fuzzy fitness. The contributions in this approach can improve the performance of existing income generating activities in alleviating user fatigue and finding optimal solutions to an optimization problem, so it is beneficial for solving complicated problems with implicit or fuzzy indices [27] we can see the user interface in figure 2.8.

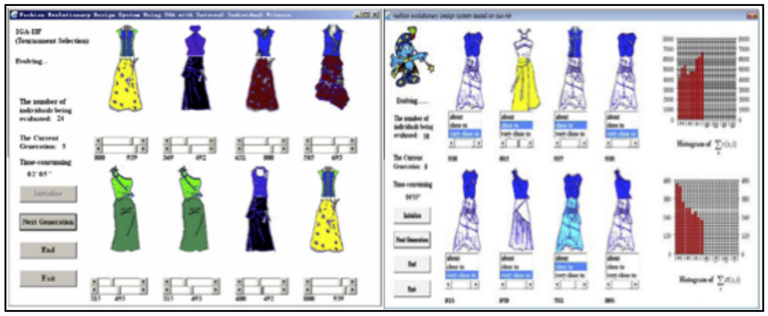

Figure 2.8: Different user interfaces interactive evolutionary system for fashion design.

2.1.1 Web-Based IEC aplications .

Picbreeder.

Picbreeder is a Web-based application that allows users to evolve images in a collaborative way maintaining a large catalog of user-created content allowing users collaboration by searching through extensive design spaces [68]. Picbreeder provides to users of all experience levels to enjoy all the creative contributions produced by other users. In this way users experience a new form of recreation called creative social recreation through collaborative exploration. In this sense these systems helps their users to find interesting images through tagging, browsing and searching as figure 2.9 show.

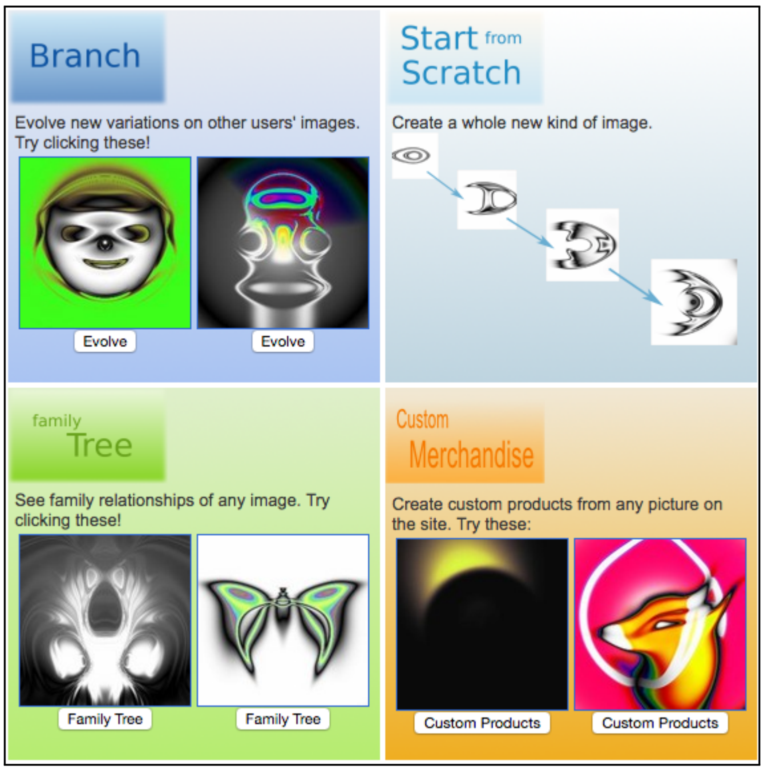

Figure 2.9: Picbreeder User Selection.

Figure 2.10: Alien objects from EndLessForms.

EndlessForms.

EndlessForms is a Web application that Explore object designs by choosing those the users like. These selected objects become the parents of the next generation of objects [13]. EndLessForms proposes a new way to evolve 3D objects inspired by biological morphologies using generative encoding. One of the experiments proposed in this paper was to use interactive evolutionary systems to determine the potential for generating complex and interesting 3D objects. They chose the interactive evolution, because that allows openended exploration of the design space of objects that can produce by their method. Additionally, the interactive evolution avoids the greedy nature of evolution by objectives, which potentially allows to access more interesting objects [13]. In figure 2.10 we can see some of this objects.

EvoSpace-Interactive

EvoSpace-Interactive is an open source framework focused on Web environments for collaborative interactive evolutionary applications. This framework defines three main components for each application, which are:

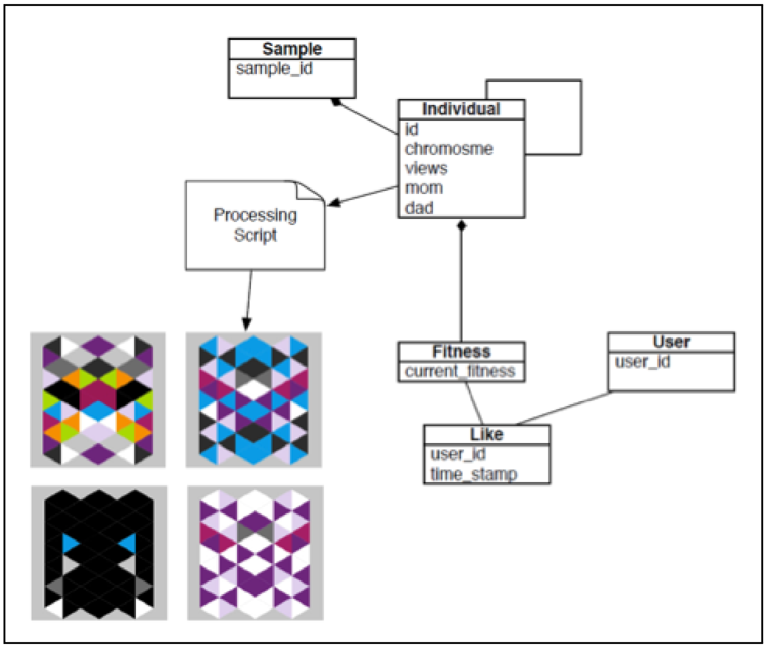

Figure 2.11: Individual Representation.

- Individual.

- Processing Script.

- Worker Script.

The individual is represented internally as a data dictionary stored in Redis [16] database management system; the individual contains main attributes as id, chromosome, mom, dad, and views. This attributes represents the key information of the individual as the individual offspring, the number of times that the individual has been selected, etcetera, as we can see in figure 2.11 [3].

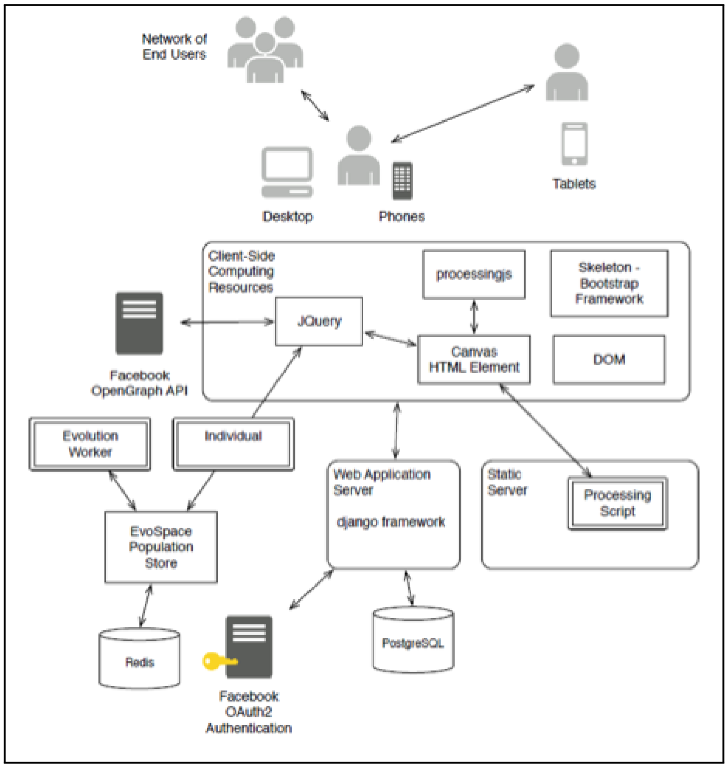

Figure 2.12: EvoSpace-Interactive Framework.

As we can observe on figure 2.12, this work uses database management systems to implement collaborative interactive evolutionary applications. One of the reasons that this framework is using Redis[15] is because it provides a hash- based implementation of sets and queues, which are natural data structures for the EvoSpace model. On the other hand this framework uses a relational database to save basic information about the user extracted from the social platform (Facebook) through open graph API and OAuth2 authentication.

2.2 User Modeling

User modeling can be represented as the technique of building a model of the user to personalize a system. The user model is commonly created as the user is working with the system. An example is an educational application that teaches students an individual skill: given the rules and knowledge in the user model, the difficulty level of the exercises in the form is altered as the user progresses. Formally definition of user modeling according to McTear [48] : " user modeling is the process of gathering information about the users of a computer system and of using the information to provide services or information adapted to the specific requirements of individual users (or groups of users)". The purpose of the user model is to have a module containing the operations that are needed to personalize the system, and the user profile, which includes the personal data of the user [](Mohamad et al., 2013). System personalization over user modeling is related to the research field of adaptive systems; this subject is beyond the scope of this research work. Focus on the human user, user modeling is a very cross-disciplinary research topic, comprehending the domains of artificial intelligence, computer science, and social science. Ideas have been coopted from an extensive range of subdomains, such as humancomputer interaction, elearning, information science, social computing, machine learning, data mining, cognitive science, and so on [37] [38]. There is interest in user modeling from both a scientific and commercial perspective [65].

2.2.1 Application Domains for user modeling

Amount Research and implementation exist in this domain in which personalization and user modeling plays an important role. This section presents several works of these domains. To understand this topic, the different objects are divided into three general categories:

- supporting a user during a task.
- giving a user a specific personalized experience.
- training and educating a user.

The categories especially differ in the kind of user data that is used. For each domain, the general purpose of the domain and the more accurate purpose of the user model are discussed.

2.2.2 User models for providing task support

Task support systems are f systems that help a user during a task by either supporting the user perform the task or by completely taking over this task [12](Nurmi et al., 2007). For instance, an application that automatically categorizes the incoming emails of the user. The goal of the user model in these requests is to promote the efficiency of interactions with the user, to simplify these interactions and to make complex systems more usable [65][25]. To perform this personalization, data is collected through observations of the user. This information is related to the users goals and needs, but especially to the task that the user currently is accomplished, like the users task knowledge and background. Much research has been done in this domain, but because many separate research projects are focusing on an exact task or subject [](Sannes, 2011), it is hard to make generalizations or to establish one delimited investigation topic. Commonly discussed research subjects are Decision Support Systems, Adaptive Hypermedia, and Adaptive Ubiquitous Systems, each having his or her own specific domain and way of personalization.

2.2.3 Decision support systems

Decision support systems are systems that support a user with making a decision in a complex, professional environment [] (Nurmi et al., 2007). For example, a system used at a pharmacy for automatically checking valid combinations of medicine. The method can be used to help the pharmacist in prescribing the right combinations and to give information for making a decision when a problem occurs. The purpose of the user model in decision support systems is to present the user with the right and appropriate information, giving different feedback or applying various decision steps according to the characteristics of the user.The data that is used is often associated with the users task and background knowledge. The adaptation takes place by adapting the amount and the content of the feedback provided by the system.

Decision support systems are traditionally ruled or logic-based systems, in which all the relevant information is represented in a knowledge base. This means that the content of the user model itself is also highly dependent on the way the rules and knowledge are represented.

2.2.4 Adaptive Hypermedia

Adaptive hypermedia system is a system that grant users to browse freely information network, structured by nodes and links, to retrieve items of information [17]. For instance an internet website application. The goal of the user model is to make the interface and structure of the system dynamic. This enables the application to adapt to the user and to make it easier for the user to search for and retrieve relevant information. The data used in the user model is related to the users abilities, knowledge, and goals in the application. The adaptation happens by adjusting the structure and the presentation style to the expected needs of the user. For example, by enhancing web search: promoting pages that might better correspond to the users characteristics, on the other hand by giving navigation support, through highlighting certain components of a page [66].

2.2.5 Adaptive ubiquitous systems

Ubiquitous systems are concerned with data handling applications integrated into everyday objects and activities (Nurmi et al., 2007). For example the smart energy meter, recording the energy usage in a household through small devices distributed in a house, supporting the user with managing this energy usage [30]. The main purpose of the user model is to improve the system, facilitate the users preferences and thus make the overall use easier. Because the personalization can take place in every situation and location, the data is focused on the user state and context. For example to enable the contextualization to a current environmental change. The adaptation takes place by changing the behavior of, and the feedback given by the whole system. These objects can be inferred by looking at the properties of the objects in the user profile, or by looking at the objects in other user profiles that are similar to the user [38] [37]. Because of the predominantly commercial goal of these systems, the adaptations often take place in a very intrusive way, to make sure the user notices the change. Most recommender systems used on the Internet, which means that some typical technical difficulties are associated with these kinds of user models. First, the user profile is often only saved during the users visit, which means that fast and efficient adaption is necessary. Recommender systems often become more precise when the user spends more time on the system. Second, the systems structure is often split up in a client and on a server side, where the client side solely gathers user information and sends it to the server, where the actual computation takes place.

2.2.6 User models for providing a personal experience.

User models for providing the user with a personal experience have the goal to improve the user experience while using the system. This kind of user modeling is especially focused on more commercial fields, such as e-commerce, marketing, and computer games, and became popular with the rise of the Internet. The information that is used by the user in this main domain is mostly focused on the information that defines the user, such as the users preferences and interests. Since this data is regularly delicate, privacy is a big issue [77]. While in other domains the privacy of the user data is also important, in this area it is even a greater topic of discussion because the incentive of the application developers is frequently contradictory to the incentive of the actual user, considering gaining and sharing the users personal information. For instance, user profiles are often shared among diverse components of the same application, or even with different applications [12] [36], which presents additional weaknesses and possible undesirable information sharing. Ensuring personal data is not open to all people, in addition to defining strict privacy policies, is thus essential in these user models. Some investigation in this domain are.

2.2.7 Recommender Systems and User Adaptive Computer Games.

Recommender systems are concerned with presenting the user with relevant information and suggestions. They are commonly used on the Internet, for example on websites such as Facebook, to provide the user with personalized news, targeted advertisements and possibly new friends [12]. The purpose of the user model is to give the system with information that is assumed to be important for the user. The information that is stored for this goal is associated with the preferences of the user to certain objects, like products, music or people. To benefit a classification of these objects, the interaction history of the user is stored, or the user is explicitly asked to rate certain objects. The content of the system is eventually adjusted by showing the recently inferred objects. In these senses, objects can be inferred by looking at the attributes of the objects in the user profile, or by looking at the objects in other user profiles that are related to the user [38] [37]. As a result of the predominantly commercial target of these systems, the adaptations often take place in a very invasive way, to make sure the user notices the change. Most recommender systems are based on the Internet, which means that some typical technical difficulties are associated with these kinds of user models. First, the user profile is often only stored while the users visit, which means that fast and efficient adaption is important. Recommender systems usually become more precise when the user spends more time with the system. Second, the systems architecture is usually client-server, where the client side gathers user information and sends it to the server, where the actual process takes place.

2.2.8 User Adaptive Computer Games.

User Adaptive computer games are games that focus on increasing the perceived value by providing a strongly individualized experience (Brisson, 2012). For example is a firstperson shooter that adapts the performance of the enemy according to the shooting accuracy of the player.

The fundamental idea of the user model is to identify or classify the user, so the appropriate adjustment is made in the computer game. The information that is used addresses the preferences and progress of the user, such as the users current difficulty level or even the employed strategy. This data is usually obtained through the interactions of the user with the game, and therefor first should be translated and formalized before it can be used to interpret conclusions on a higher level. The adaptation that takes place in the game concerns changing the content and role of the game, such as the game difficulty, the behavior of nonplayer characters or even the background music [3].

Because of the emphasis on the user, user adaptive computer games have relatively a lot of processing power available for personalization. In this sense the user adaptive computer games domain is a very interesting research domain.

2.2.9 User models for educational purposes.

Educational systems are systems developed with a teaching reason. They are commonly applied in elearning, where electronic media and Information technologies are used for education. However, in most educational systems, user modeling and adaptation plays a minor role. Content is presented, and only simple things such as the students progress in the course are registered. By adding personalization to these applications, the learning value can notably increase, ensuring that every learner achieves and reaches the highest standards possible [31]. Also, the experience of the teacher or supervisor can be increased through personalization, for instance through inferring and employing the preferred teaching style. However, here we consider the student as the user to which the system will be personalized. Thus, it is preferable that the data stored in the user profile be interpretable by humans. When looking at the time of adjustment in educational systems, we can make a clear difference between adaptation while the student is doing an exercise, which we will refer to as online adaptation, and adaptation that takes place afterward, which we will refer to as offline adaptation. The most important investigation domains that do utilize considerable user modeling constructions are Intelligent Tutoring Systems and

Adaptive Educational Games.

2.2.10 Intelligent Tutoring Systems.

Intelligent Tutoring Systems (ITS) are systems that provide students automated stepbystep instruction as the students complete training tasks and/or work on exercises. An ITS has the purpose to complement or even replace the human teacher. For example a system for teaching students how to program, with the ability to automatically detect common mistakes [23].

The particular goal of the user model is to select educational activities and strategies and in addition delivered individual feedback that is most relevant to the users level of knowledge [38] [48]. The user information that is stored for this purpose is the students state, knowledge and level of achievement. This data is exclusively observed over the actions and results of the student, such as the answers the student gives. After observing this information, it is used to infer higher level properties, such as the students learning style and other preferences.

Traditionally, just like decision support systems, information technology systems are knowledge based systems, using formalized domain knowledge and rules to drive the user adjustment (adaptation). For instance stereotypes are widely used in information technology systems [37] and represent a set of default attributes that often cooccur in users or in a certain group of people. The different stereotypes that have been build differ in granularity of detail and complexity.

2.2.11 Adaptive Educational Games.

Adaptive Educational Games (AEGs) are complicated educational games that combine ideas from several investigations areas, to increase the students learning experience [62]. These are especially based on serious games: computer games with an educational approach, where things are taught to students by using a playful idea [39] [35]. For instance an AEG is a training application for fire fighters, letting the fire fighters train their skills and knowledge in a safe on a virtual environment.

The objective of the user model in an AEGs is to optimize the learning process and outcome. The user information is considered with the advance and knowledge of the student, but also with the students mental and cognitive characteristics. The gained data can be used to adjust the content, presentation, and system behavior to the students need, for example, by adjusting the content, tone, or amount of presented feedback. Adaptive computer games have a lot of processing power available for personalization, making a complex and interesting domain for user modeling.

2.2.12 Methods for user modeling.

In the user modeling topic, researchers have proposed more general design methods and frameworks to guide the developers in the process of user modeling. These general methods are useful in research projects, where the knowledge can be reused to adjust the user model to the systems characteristics. Also in commercial applications, these general methods have proven to be useful [12] , because they make it easier and more feasible to implement personalization into a system. In early work, the process of user modeling was mostly based on the intuition and experience of the developer or researcher. In recent work, the techniques of user modeling were essentially based on the intuition and expertise of the developer or researcher. As the user modeling research field evolved, there has been put much effort in creating a general way for designing and constructing a user model, by basing decisions on more empirical grounds and by defining methods applicable to the whole field [38] [22].

Frameworks, methodologies, and architectures have been developed, defining the strict process, restrictions and choices on how to design and build a user model. In the early days of user modeling, the focus was put on developing one method applicable to the user modeling field as a whole. However, user modeling is a very cross-disciplinary research subject. Therefore, throughout the decades, the user modeling area of research has been influenced by the important research topics and trends of their time. For example, when information technologies became a major subject in the early nineties, user modeling methods were also mostly focused on the application of stereotypes, knowledge bases, and logic to define a user model. With the rise of the Internet, the objectives of the user modeling field change to Web-oriented applications and all the specific problems that arise with this. Thus this connection, also the general user modeling methods that were developed, were focused on the popular research domains of their time [37]. The main approaches to user modeling did not change, but the specific fillingin of the user model, such as which technology to apply, did change. In this sense development of user modeling as a whole, most researchers eventually agreed that one method to solve all problems is not possible [48] [38]. Instead, a broad range of generic user modeling methods has been developed [25]; each of which supports only a few of the very different manifestations of personalization.

2.3 Gamification.

A definition given by Huotari [34] is Gamification is the process by which concepts are brought to the real world task associated with real people. Also gamification handle game design elements which are commonly known as non-game context in the presud to enhance user engagement, organizational productivity, flow, learning, evaluations, among others.

Games and game technologies increase exponentially the traditional boundaries of their medium, as evidenced by the growth of serious and pervasive games as an industry and research field. The most recent phenomenon in this path is gamification, paradigm for the use of video game elements (rather than full- fledged games) to improve user experience and user engagement in non-game services and applications [20].

2.3.1 Techniques.

Techniques in this context seek to persuade users to use their natural desire to compete, learn and socialize in given non-game context application [19][29]. Some works in the beginning used rewards for users as players to perform desired tasks in a certain application or involving users to compete with each other . For instance some sort of rewards include points [73], achievement badges or levels [28], the filling of a progress bar [54], or providing the user with virtual currency. By Making the rewards for tasks achievements visible to other players or providing leader boards are ways of encouraging players to compete [32]. Because the problematic consequences of competition, which can result in negative conduct, low cooperation and collaboration, or disadvantaging certain player demographics such as women [40], best-practice gamification designs try to refrain from using this element.

Another techniques to gamification is to make existing tasks feel more like games [18]. Some techniques used in this approach include adding meaningful choice, onboarding with a tutorial, increasing challenge, and adding narrative [46].

Chapter 3

Background

This chapter present the fundamental concepts related to this work. The formal definitions referring to fuzzy systems, user modeling concepts and gamification theory and techniques related to this method.

3.1 Fuzzy Logic.

Fuzzy logic is an approach to computing based on "degrees of truth" rather than the usual "true or false" (1 or 0) Boolean logic on which the modern computer is based. The idea of fuzzy logic was first advanced by Dr. Lotfi Zadeh of the University of California at Berkeley in the 1960s. Dr. Zadeh was working on the problem of computer understanding of natural language. Natural language (like most other activities in life and indeed the universe) is not easily translated into the absolute terms of 0 and 1. (Whether everything is ultimately describable in binary terms is a philosophical question worth pursuing, but in practice much data we might want to feed a computer is in some state in between and so, frequently, are the results of computing.) It may help to see fuzzy logic as the way reasoning works, and binary or Boolean logic is simply a special case of it. Fuzzy logic includes 0 and 1 as extreme cases of truth (or "the state of matters" or "fact") but also includes the various states of truth in between so that, for example, the result of a comparison between two things could be not "tall" or "short" but ".38 of tallness."

Fuzzy logic seems closer to the way our brains work. We aggregate data and create some partial truths which we aggregate further into higher truths which in turn when certain thresholds are exceeded, cause certain further results such as motor reaction. A similar kind of process is used in neural networks, expert systems, and other artificial intelligence applications. Fuzzy logic is essential to the development of human-like capabilities for AI, sometimes referred to as artificial general intelligence: the representation of generalized human cognitive abilities in software so that, faced with an unfamiliar task, the AI system could find a solution.

3.1.1 Fuzzy inference system.

A fuzzy inference system (FIS) is a system that uses fuzzy set theory to map inputs (features in the case of fuzzy classification) to outputs (classes in the event of fuzzy classification). An example of a Mamdani inference system is shown in Figure x To compute the output of this FIS given the inputs; one must go through six steps:

1. **Determining a set of fuzzy rules.**

2. **fuzzifying the inputs using the input membership functions.**

3. **Combining the fuzzified inputs according to the fuzzy rules to establish a rule strength.**

4. **Combining the fuzzified inputs according to the fuzzy rules to establish a rule strength.**

5. **Combining the consequences to get an output distribution.**

6. **Defuzzifying the output distribution (this step is only if a crisp production (class) is needed).**

The following is a more detailed description of this process.

3.1.2 Creating fuzzy rules.

Fuzzy rules are a collection of linguistic statements that describe how the FIS should make a decision regarding classifying an input or controlling an output. Fuzzy rules are always written in the following form: if (input1 is membership function1) and/or (input2 is membership function2) and/or . then (output is output membership function). For example, one could make up a rule that says: if temperature is high and humidity is high then room is hot. There would have to be membership functions that define what we mean by high temperature (input1), high humidity (input2) and a hot room (output1). This process of taking an input such as temperature and processing it through a membership function to determine what we mean by "high" temperature is called fuzzification and is discussed in section 3.1.2. Also, we must define what we mean by "and" / "or" in the fuzzy rule. This is called fuzzy combination and is discussed in section 3.1.3.

3.1.3 Fuzzification.

The purpose of fuzzification is to map the inputs from a set of sensors (or features of those sensors such as amplitude or spectrum) to values from 0 to 1 using a set of input membership functions. In the example shown in figure 3.1, there are two inputs, x0, and y0 is shown in the lower left corner.These inputs are mapped

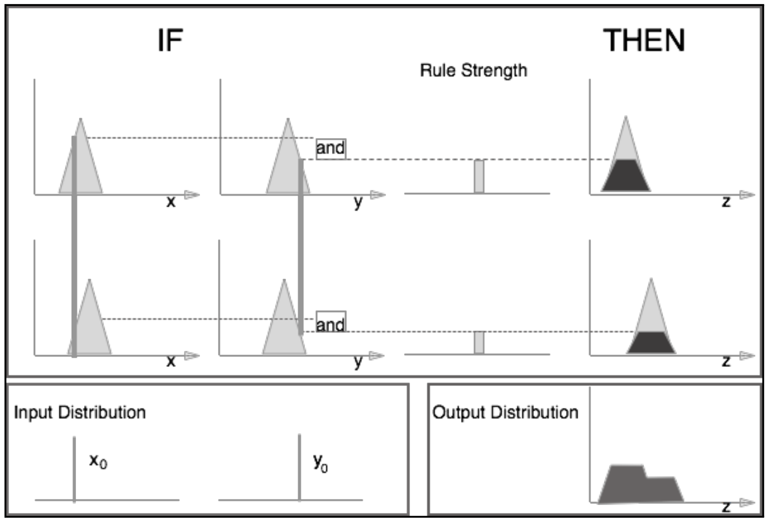

Figure 3.1: A two input, two rule Mamdani FIS with crisp inputs.

into fuzzy numbers by drawing a line up from the inputs to the input membership functions above and marking the intersection point.

These input membership functions, as discussed previously, can represent fuzzy concepts such as ”large” or ”small”, ”old” or ”young”, ”hot” or ”cold”, etc. For example, x0 could be the EMG energy coming from the front of the forearm and y0 could be the EMG energy coming from the back of the forearm. The membership functions could then represent ”large” amounts of tension coming from a muscle or ”small” amounts of tension. When choosing the input membership functions, the definition of what we mean by ”large” and ”small” may be different for each input.

3.1.4 Fuzzy combinations (T-norms).

In making a fuzzy rule, we use the concept of "and", "or", and sometimes "not". The sections below describe the most common definitions of these "fuzzy combination" operators. Fuzzy combinations are also referred to as "T-norms".

3.1.5 Fuzzy "and"

The fuzzy "and" is written as:

$$\mu_A \cap \mu_B = T(\mu_A(x), \mu_B(x)) \tag{3.1}$$

where A is read as "the membership in class A" and B is read as "the membership in class B". There are many ways to compute "and". The two most common are:

1. Zadeh - $min(\mu_A(x), \mu_B(x))$. This technique, named after the inventor of fuzzy set theory simply computes the "and" by taking the minimum of the two (or more) membership values. This is the most common definition of the fuzzy "and".

2. Product - $\mu_A(x) + \mu_B(x)) - \mu_A(x)\mu_B(x))$. This technique uses the difference between the sum of the two (or more) membership values and the product of the membership values.

Table 3.1: The Boolean ”or”.

Input 1	Input 2	Input 3
0	0	0
0	1	1
1	0	1
1	1	1

Both techniques have the following properties:

- $T(a, 0) = T(0, a) = a$
- $T(a, 1) = T(1, a) = 1$

Similar to the fuzzy ”and”, both definitions of the fuzzy ”or” also can be used to compute the Boolean ”or”. Table 3.1 shows the Boolean ”or” operation. Notice that both fuzzy ”or” definitions also work for these numbers. The fuzzy ”or” is an extension of the Boolean ”or” to numbers that are not just 0 or 1, but between 0 and 1.

3.1.6 Fuzzy ”and”

The consequence of a fuzzy rule is computed using two steps: 1. Computing the rule strength by combining the fuzzified inputs using the fuzzy combination process discussed in section 4.1.3. This is shown in Figure 4-1. Notice in this example, the fuzzy ”and” is used to combine the membership functions to compute the rule strength. 2. Clipping the output membership function at the rule strength. Once again, refer to Figure 4-1 to see how this is done for a two input, two rule Mamdani FIS.

3.1.7 Combining Outputs into an Output Distribution

The outputs of all of the fuzzy rules must now be combined to obtain one fuzzy output distribution. This is usually, but not always, done by using the fuzzy ”or”. Figure 4-1 shows an example of this. The output membership functions on the right-hand side of the figure are combined using the fuzzy ”or” to obtain the output distribution shown in the lower right corner of the figure.

3.1.8 Defuzzification of Output Distribution

In many instances, it is desired to come up with a single crisp output from a FIS. For example, if one was trying to classify a letter drawn by hand on a drawing tablet, ultimately the FIS would have to come up with a crisp number to tell the computer which letter was drawn. This crisp number is obtained in a process known as defuzzification. There are two common techniques for defuzzifying:

1. Center of mass - This technique takes the output distribution found in section 4.1.5 and finds its center of mass to come up with one crisp number. This is

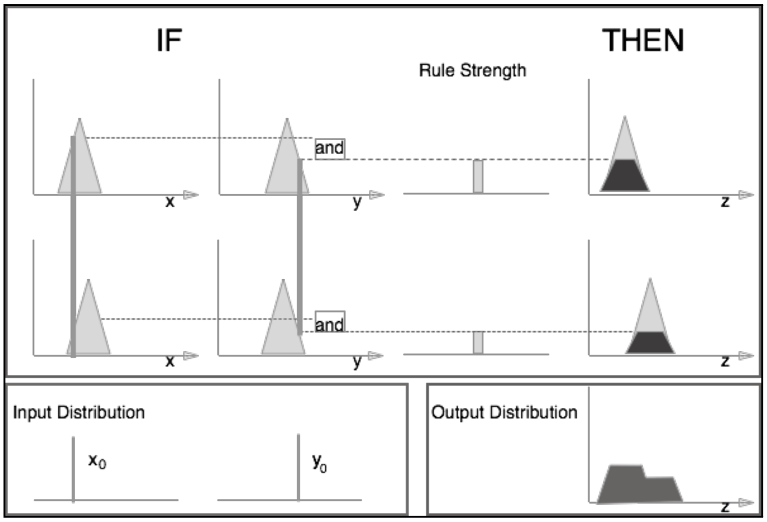

Figure 3.2: Defuzzification Using the Center of Mass.

computed as follows:

$$z = \frac{\sum_{j=1}^{q} z_j \mu_c(z_j)}{\sum_{j=1}^{q} \mu_c(z_j)} \tag{3.2}$$

where z is the center of mass and uc is the membership in class c at value zj. An example outcome of this computation is shown in Figure 4-2.

Falta agregar imagen al path

2. Mean of maximum - This technique takes the output distribution found in section 4.1.5 and finds its mean of maxima to come up with one crisp number. This is computed as follows:

$$z = \frac{\sum_{j=1}^{l} z_j}{l} \tag{3.3}$$

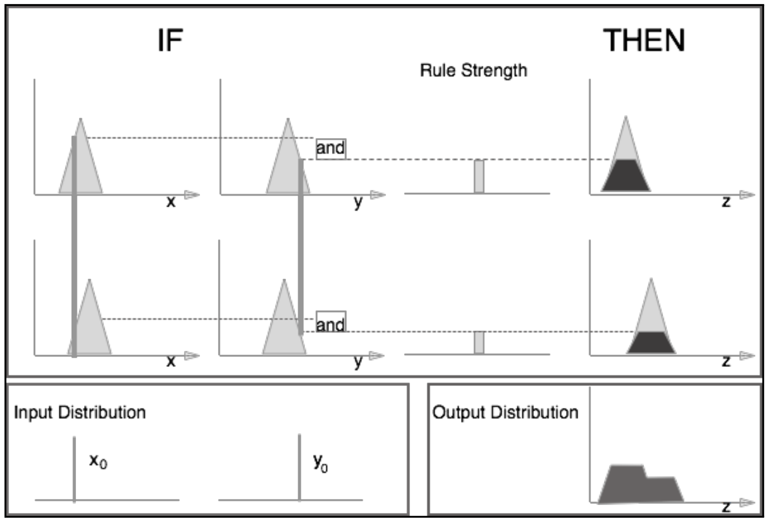

Figure 3.3: Defuzzification Using the Mean of Maximum.

where z is the mean of maximum, zj is the point at which the membership function is maximum, and l is the number of times the output distribution reaches the maximum level. An example outcome of this computation is shown in Figure x.

Falta agregar imagen al path

3.1.9 Fuzzy Inputs.

In summary, Figure x-1 shows a two input Mamdani FIS with two rules. It fuzzifies the two inputs by finding the intersection of the crisp input value with the input membership function. It uses the minimum operator to compute the fuzzy "and" for combining the two fuzzified inputs to obtain a rule strength. It clips the output

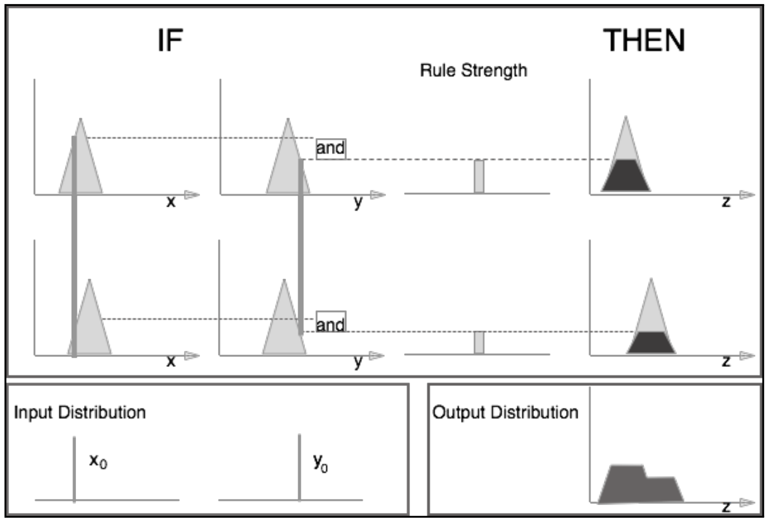

Figure 3.4: A two Input, two rule Mamdani FIS with a fuzzy output.

membership function at the rule strength. Finally, it uses the maximum operator to compute the fuzzy "or" for combining the outputs of the two rules. Figure x-4 shows a modification of the Mamdani FIS where the input y0 is fuzzy, not crisp. This can be used to model inaccuracies in the measurement. For example, we may be measuring the output of a pressure sensor. Even with the exact same pressure applied, the sensor is measured to have slightly different voltages. The fuzzy input membership function models this uncertainty. The input fuzzy function is combined with the rule input membership function by using the fuzzy "and" as shown in Figure x-4.

falta agregar el path de la imgen

Chapter 4

Proposed user-centered framework

The fundamental goal of this research is to develop a user-centered framework for interactive evolutionary computation (IEC) in order to increase users participation and also to minimize the amount of evaluations needed for the evolutionary process in given Web-based IEC application.

In this chapter an explanation of the proposed framework is described. The different techniques used, such as user modeling, fuzzy logic, and human- interaction.

This framework is presented in figure 4.1. Each of the entities of this framework will be explained in detail in following sections.

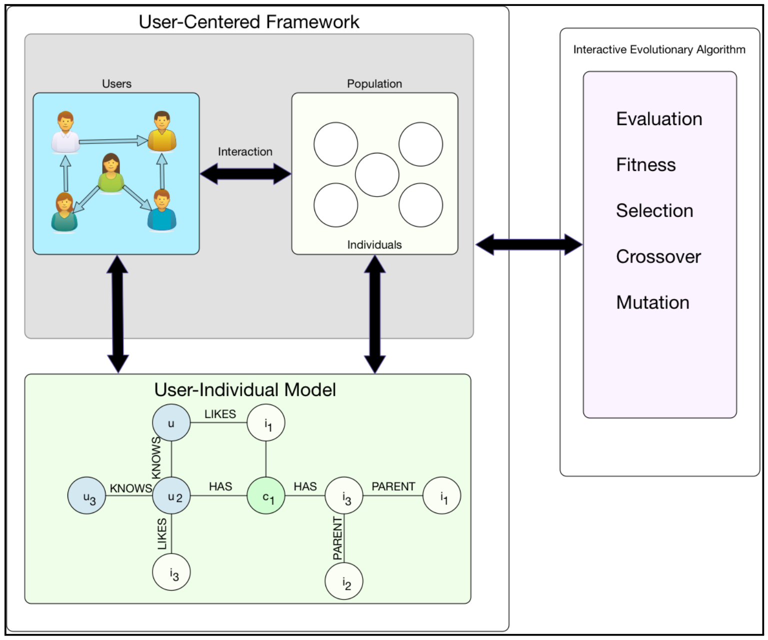

Figure 4.1: User-Centerd Framework.

4.1 Users

Users are a central part of this proposed framework as it aims to study their behavior when interacting with individuals within interactive evolutionary algorithms and other tasks that may exist within IEC systems. Thus users in this proposal are entities interacting with an evolutionary computation that has the fundamental purpose of evaluating individuals of a population replacing the fitness function according to their preferences and mood, among others []. This form of evaluate individual is given a subjectively way.

On the other hand in order to capture the attention of users and possibly increase their participation, currently there exist some basic actions that users can perform

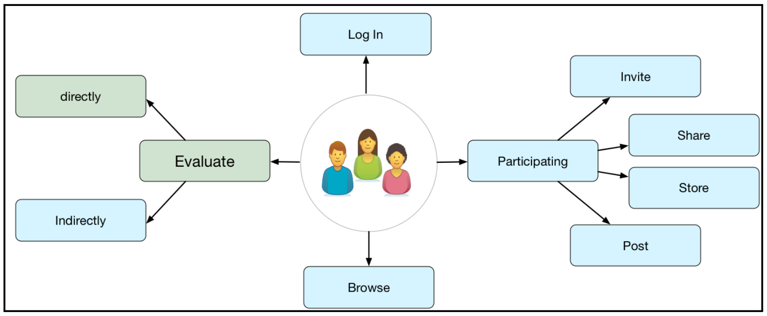

Figure 4.2: Users actions on interactive evolutionary systems.

in a given system, going from to be able to access a system through login mechanism, once the user has logged-in in to the system the users can interact with different actions, for example the inviting action is when users can be invited through social networks or maybe of person to person. Another example is the sharing action, that can occur when users want to share something of their interest through their social networks. In this sense the action of posting is when users want to put something on their wall who want others to know through their social networks. Likewise the storing action occurs when users want to retain permanently information that are of their interest. Finally the browsing action occurs when users are exploring content for their needs. All these actions go beyond only evaluating individuals.

In order to start the task of evaluating individuals is necessary that users access the system through a login mechanism. This mechanism consists of providing a username and a password in order to grant access to the system. All users accessing in this way they are considered active users within an IEC system.

For the evaluating action is proposed that users evaluate individuals indirectly, this means users can evaluate accepting indirect recommendations of friends that are currently active in the system. These recommendations can be store individuals in the system of users who know each other.

Also the browsing action is proposed within IEC systems, this means that users can explore information that other active users in the system are generating.

Finally activity participation is proposed. This can be divided into four different actions as fallows:

- The Inviting action occurs when a user invites another to the system through social networks or from person to person.

- The sharing action occurs when users share their individual creations with others active user in the system.

- The posting action occurs when published what they are doing within the system.

- The storing action occurs when users keep individuals in a collection, the collections concept will be discussed later in this chapter.

All the mention above is shown in figure 4.2

Knowing these factors a graph-based user modeling is developed. Where the vertices are defined by set of users, individuals, and collections. The Edges are define by set of relationships as LIKES, KNOWS, HAS, PARENT that represents the relationships between the vertices as seen in 4.3. In this sense a formal definition is given by following definition.

$$V = \{[u_1, u_2, u_3, ..., u_n], [i_1, i_2, i_3, ..., i_n], [c_1, c_2, c_3, ..., c_n]\},$$

$$E = \{[l_1, l_2, l_3, .., l_n], [p_1, p_2, p_3, ..., p_n], [h_1, h_2, h_3, ..., h_n], [k_1, k_2, k_3, ..., k_n]\}$$

Where V is a set of vertices and u represents users, i represents individuals, and c represents collections. Also E is a set of edges where l represents the relationship "LIKES" ,p represents the relationship "PARENT", h represents the relationship "HAS" and finally k that represents the relationship "KNOWS".

In each vertex is necessary to store some knowledge about them, for instance in the vertex u has the properties identifier, name, creation date and name type. Where identifier is a unique number to identifies the user, creation date is a property where the vertex was created and name type is label for classify the vertex in this case a Person label is apply. Also for vertex i has the properties identifier, creation date, name type. Where identifier is a unique alphanumerical value that identify the individual, creation date is when this vertex is created and finally name type that is a label of classify the vertex in this case a Individual label is apply. Now for the c vertex has the same properties as the u vertex for their definitions of identifier and creation date, the difference is the name type property that is a table for classify the vertex in this case a Collection label is apply.

It is important to mention that the application for this research is using EvoSpace-Interactive through the EvoSpace framework. Also another thing to mention is the generation of the initial individual population process in EvoSpace is gather to transform this individuals in vertices or nodes in order to be ready for the evaluations of the users.

This user modeling is created when the users starts to interact whith the application. In order to a user start evaluating individuals in the application an essential process is required. These process is that the user has a Facebook account. Ones the user is access he/she can starts to interact with the different tasks presented by the application. Thus the user can evaluate, generate collection of individuals and also views friends individuals collections presented in the human-interacting interface, bring us the creation of the graph-based user modeling.

4.2 Individual

Individuals are entities that form the population for the evolutionary algorithm. In particular for this work individuals are animated digital paintings, every individual in the population is defined by a chromosome []. This chromosome defines the

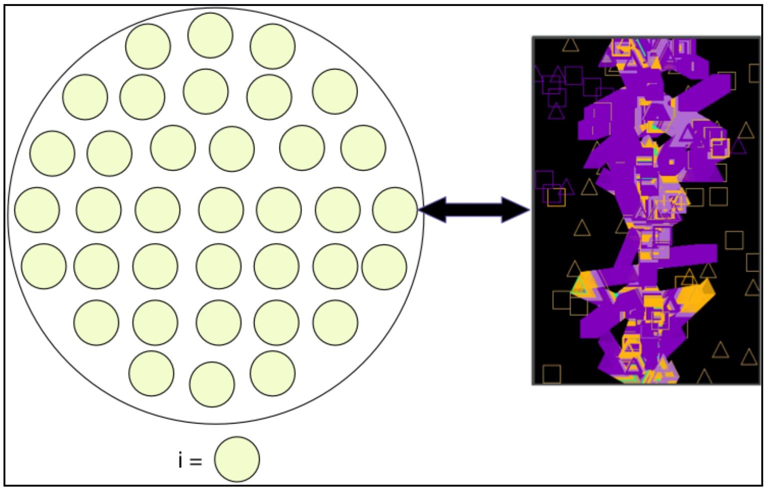

Figure 4.3: Individual representation.

behavior of the individual, so that it consists of a vector of real numbers of fifteen genes, where each gen define a particular behavior in painting. This individuals combined each other to generated new individual in the population following the genetic operators such as selection, crossover and mutation[]. An example of a individual in this case it is illustrated in figure **??**

etc. In figure 4.3 we can observe a chromosome composed of 15 elements from a digital paint. Once the individual is shown, the users can proceed to evaluate the individual subjectively. This means that according to the user's preferences the user can print his taste in all the evaluations. The evaluation method consists of giving the desired rating through an interactive visual component of five stars. The interactive visual component allows the user to select from one to five stars to evaluate the individual, having the one star as a slightly liking rate and the five stars like a total satisfaction rate. The application enables the user to visualize his social network and the personal evaluation of the users from his social net. In addition to this, the users can create collections with the main goal to allow the users to stock the more pleasing digital paints based on his preferences. Also, the user interface contains a section called About where the application explains to users in a general way what the application is all about. For users interact with the interactive evolutionary computation system it was necessary to develop an application which is called EvoDrawings. In this point an explanation of the human-computer interface that was used is given.

EvoDrawing It is a Web-based application of interactive evolutionary computation. In this application users need an account of the social network platform in particular "Facebook" to access the application and consequently participate. Once users access can evaluate individuals presented as a digital painting that is composed of a chromosome, where its representation is a vector of real numbers of fifteen positions. Each position in the chromosome represents some kind of behavior, for instance the type of figure, movement, color, and so on. Figure 4.4 chromosome composed of 15 items represent a digital painting is displayed. Once the individual is shown, users come to evaluate in a subjectively way. This means that according to the likes, preferences, and so on., users conduct their evaluations.

To evaluate individuals, users are presented with a component of visual interac-

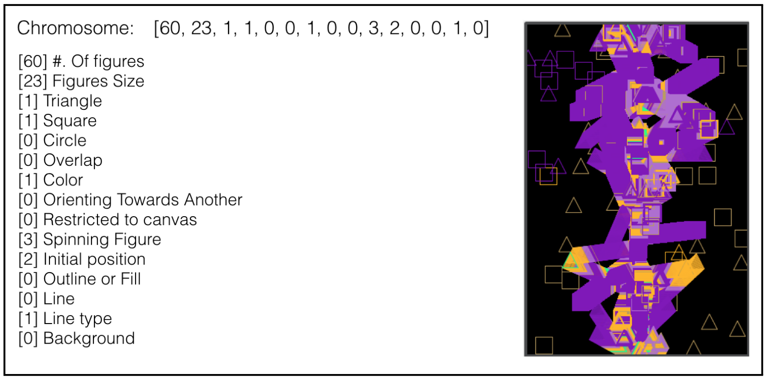

Figure 4.4: CHROMOSOME REPRESENTATION.

tion based on star ratings. This visual component is from one star to five star where a star means that the user does not like much the painting and five means total satisfaction. Within the applications users can create collections in order to store paintings are met their expectations, in other words having a grate like for them.

Users can view their friends who are using the application as well as see what level of experience (participation) in a leaderboard. This in order to motivate them to evaluate more Individuals and consequently Increase Their participation Within the application.

In the interface there is a section of "About" where he explains to users in a general way that addresses the application of EvoDrawigns. In the interface there is a section of "About" which explains to the users in a breve way what the application is all about. This human-computer web-based interface can be seen in figure x.

We can find the vertices that are defined by the users, individuals, and collections as we can observe in figure 1. The users has the following properties:

4.3 User activity

The user activity is not so different from how the graph is formed, with the difference that the information generated is formed using the standard specified by "JSON activity stream 1.0," which is stored in the engine database NoSQL "Redis ."

An activity consists of 4 elements: an actor, a verb, an object and a target. The activity generates the history of a user and performs an action on an object. For example - "Christian likes the individual 3" or "Mario created a collection". In most cases, the components will be explicit, but may also be implicit.

The primary goal of this specification is to provide sufficient metadata about the activity, to help the consumer of these data to present the information to the user, in a straightforward and user-friendly format. This involves building simple sentences on the activity that is happening, as well as the visual representation of the activity.

So far, we can say that an "Activity Stream" is a collection of one or more activities of an individual (user). This specification does not define the relations between the activity within the collection, therefore remains to the interpretation of the user who implements it.

4.4 Database for interactive evolutionary computation

Our data store for interactive evolutionary computation consists of two databases, as figure **??** shown. On one hand one data base engine for EvoSpace uses Redis database already explained in Chapter x. This database contains the structure of individual-user data, where every user participation is store, for instance the fitness of the individual, the user identifier, the representation of the chromosome, and so on. It is necessary to have this information because is used in the fuzzy inference block , which is explained later in this chapter. On the other hand there is a relational database, where basic user information is stored, such as a user ID, your email, user session. Also this database stores everything needed to meet the requirements process login "Facebook". It also contains the structure for storing collections as shown in Figure reffig:databases.

The database for interactive evolutionary computation consists of two databases, as we observe in figure X. In one hand we have the EvoSpace database that uses Redis database engine that is already explained in Chapter X section X.

This database contains the structure of individual user data, where every user participation is being stored as well as the fitness of the individual, the user identifier, the representation of the chromosome, etc.

It is necessary to have this information because it is used for fuzzy inference block, which is explained in another chapter. On the other hand, we have a relational database, where basic user information is stored, such the ID, the email, the session, etc. This database also stores everything needed to meet the requirements of login for "Facebook". One important additional information is that contains the structure for storing collections as we can see in Figure X.

4.5 Fuzzy Inference

In this chapter, we will focus on the explanation of fuzzy inference block. This block uses fuzzy inference to acquire a parameter having the weight function to adjust the participation in interactive evolutionary computing applications. This parameter is used in the decision block which will be explained later in this chapter. Additionally, the parameter is also acquired from the information generated in different databases.

The fuzzy inference system is Mamdani type and is composed of three inputs and one output. Where entries are defined by the preference variable that is also composed of three functions of triangular membership; these features are called low, medium, high, and have ranged from 1 to 5. This range is given by the preference that the user assigns to the individual at the time of assessment.

We also consider the input variable called "experience" and it is defined by three functions of triangular membership under the name of low, medium and high in a range of 1 to 100. The range of this variable are acquired from the activities that the user performs in the application, and empirically with each activity, a score is assigned. For example, if the user makes a login, a three-point score is assigned, as well if the user evaluates an individual, a two-point score is assigned, all the activities has a score punctuation and also the user has a score limit of one hundred points.

The third variable which is called ranking it is also defined by three triangular membership functions with the name of low, medium, high, in a range of 1 to 30. This range is defined by a ranking process as well as is also performed in video games; the range is adjusted according to the user participation.

This involvement is acquired from all the cardinality of the graph that the user has and passes through the logarithmic equation X that calculates the value of ranking. Finally, the exit ”fuzzyrate” is in a range of 1 to 100 defined by three triangular membership functions with the name of bad, normal and good. In figure reffig:Fuzzy shows this fuzzy inference system.

1. *If **preference** is low and **experience** is low and **ranking** is low then **fuzzy-rate** is bad.*

2. *If **preference** is low and **experience** is low and **ranking** is mid then **fuzzy-***

rate *is bad.*

3. *If **preference** is low and **experience** is low and **ranking** is high then **fuzzy-rate** is bad.*

4. *If **preference** is low and **experience** is mid and **ranking** is low then **fuzzy-rate** is bad.*

5. *If **preference** is low and **experience** is mid and **ranking** is mid then **fuzzy-rate** is bad.*

6. *If **preference** is low and **experience** is mid and **ranking** is high then **fuzzy-rate** is normal.*

7. *If **preference** is low and **experience** is high and **ranking** is low then **fuzzy-rate** is normal.*

8. *If **preference** is low and **experience** is high and **ranking** is mid then **fuzzy-rate** is normal.*

9. *If **preference** is low and **experience** is high and **ranking** is high then **fuzzy-rate** is normal.*

10. *If **preference** is mid and **experience** is low and **ranking** is low then **fuzzy-rate** is bad.*

11. *If* ***preference*** *is mid and* ***experience*** *is low and* ***ranking*** *is mid then* ***fuzzy-rate*** *is normal.*

12. *If* ***preference*** *is mid and* ***experience*** *is low and* ***ranking*** *is high then* ***fuzzy-rate*** *is normal.*

13. *If* ***preference*** *is mid and* ***experience*** *is mid and* ***ranking*** *is low then* ***fuzzy-rate*** *is normal.*

14. *If* ***preference*** *is mid and* ***experience*** *is mid and* ***ranking*** *is mid then* ***fuzzy-rate*** *is normal.*

15. *If* ***preference*** *is mid and* ***experience*** *is mid and* ***ranking*** *is high then* ***fuzzy-rate*** *is normal.*

16. *If* ***preference*** *is mid and* ***experience*** *is high and* ***ranking*** *is low then* ***fuzzy-rate*** *is normal.*

17. *If* ***preference*** *is mid and* ***experience*** *is high and* ***ranking*** *is mid then* ***fuzzy-rate*** *is normal.*

18. *If* ***preference*** *is mid and* ***experience*** *is high and* ***ranking*** *is high then* ***fuzzy-rate*** *is normal.*

19. *If* ***preference*** *is high and* ***experience*** *is low and* ***ranking*** *is low then* ***fuzzy-rate*** *is normal.*

20. *If **preference** is high and **experience** is low and **ranking** is mid then **fuzzy-rate** is normal.*

21. *If **preference** is high and **experience** is low and **ranking** is high then **fuzzy-rate** is good.*

22. *If **preference** is high and **experience** is mid and **ranking** is low then **fuzzy-rate** is normal.*

23. *If **preference** is high and **experience** is mid and **ranking** is mid then **fuzzy-rate** is normal.*

24. *If **preference** is high and **experience** is mid and **ranking** is high then **fuzzy-rate** is good.*

25. *If **preference** is high and **experience** is high and **ranking** is low then **fuzzy-rate** is good.*

26. *If **preference** is high and **experience** is high and **ranking** is mid then **fuzzy-rate** is good.*

27. *If **preference** is high and **experience** is high and **ranking** is high then **fuzzy-rate** is good.*

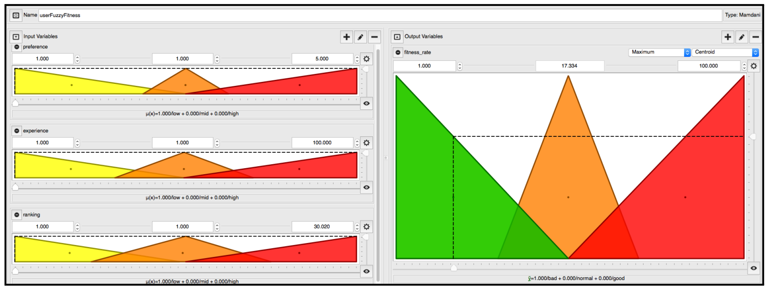

Figure 4.5: Fuzzy Inference System.

4.6 Decision Making

The decision block it is defined by equation 4.1 representing the value of fitness that takes the individual to be evaluated by the user, as well as everything else that makes in the application.

$$f(ff) = \frac{\sum_{i=0}^{n} x_i f(y_i)}{\sum_{i=0}^{n} f(y_i)} \tag{4.1}$$

Where ff represents the fuzzy fitness function for all users who have evaluated an individual of the population in particular. Likewise x represents the range that a user assigned to the individual according to their preferences, likes, and so on. The function $f(y_i)$ represents the fuzzy inference and is defined by the equation 4.2.

$$f(y_i) = fr(x, e, r) \tag{4.2}$$

Where x is still the range that users assigned to a particular individual. The variable $e$represents the user experience, which is a function that is defined in equation 4.2.

$$f(e) = \sum_{i=0}^{n} l_i j_i s_i s_i \tag{4.3}$$

Where i represents user activity. The variable l represents the taste generated of the verb "like" from the user activity stream. The variable j represents the access that like the variable is obtained from the verb "join" from the user activity stream. The variables s, o are represented by the verbs "save" and "open" from the user activity stream.

$$s = \frac{levels}{log_2 fp} \tag{4.4}$$

In equation 4.2, we can find the variable r as the last parameter of the function and represents a ranking of the user; this variable is defined by a logarithmic scale where intervenes our graph-base user modeling and is defined by Equation 4.4.

In equation 4.4, the variable s represents the scale and $levels$ represents the highest level that the users can have. The variable fp represents the final points, which are the maximum points that a particular user can have.

In order to acquire the user's ranking level , a floor function is calculated, the main point of this is to increase the difficulty for the high ranked users to level up.

In other words the expert users needs to have more participation if they want to increase their level ranking. This function is defined by equation 4.5.

$$rl = \lfloor (s)(log_2 pf) \rfloor \tag{4.5}$$

Where rl represents the ranking level, s represents the scale and are all the participations that the user has made. These participations represents the degree that the user has within his own graph and is defined by the vicinity of its vertex that is given by the adjacent vertices to u, defined in Equation 4.6.

$$N(u) = \{y \in V_G | \{u, y \in E_G\} \tag{4.6}$$

In this case the degree of the vertex u is the number of neighbors of:

$$g(u) = |N(x)| \tag{4.7}$$

Chapter 5

Case study

This chapter will explain the experiment used in the Study Case. Three versions of EvoDrawing were used to fulfill this research.

Starting off with the hypothesis which will allow increasing the user experience of having a drawing modeling using diffuse logic and use techniques.

It was necessary to design Three experiments to prove this thesis work hypothesis; each experiment used a different version of EvoDrawing. The results as well as each version of EvoDrawing compared to other applications of interactive evolutionary computing and also were compared between other Evo Drawing versions itself. Equality of data, parameters, and users compared to all experiments which examined as follows:

It is necessary to compare our proposal with other which we have explained early in the last chapter, due evolutionary interactive computing applications with the same amount of parameters are difficult to find we decided to build two additional ones in order to run pertinent tests so our hypothesis could be answered., This required to design experiments the following way.

The X table contains Evo Drawing versions and comparison. The first column will specify each version of Evo Drawing including its characteristics and each subsequent column describes the characteristics of each Evo Drawing version.

Now we will explain each version developed details and goes as follows; we will explain each version specific and technical reasons.

5.1 EvoDrawings01

This version initial configuration of our evolutionary algorithm goes as follows:

1. **Having an initial 80 individual population which we will represent in the following way:**

2. **evaluations equals an evolution parameter.**

3. **Its fitness provided by equation x.**

4. **Combining the fuzzified inputs according to the fuzzy rules to establish a rule strength.**

Table 5.1: comparison of the different versions.

Version	EvoDrawing01	EvoDrawing02	EvoDrawing03
User Interface	This version have an easy to handle and simple gui.Observe figure X	This version have an easy to handle and simple gui. Observe figure X	As the previous versions this one have the same interface but differs from the others on the competitiveness visual components based on video games and we can observe it in figure X
Fuzzy Inferance	It does not have Fuzzy inference system	A fuzzy inference system has been added to this version. It is composed by two inputs Experience and Preference; it also has a type Mamdani output called fuzzy rate.	A fuzzy inference system has been added to this version. It is composed by three inputs Ranking, Experience and Preference; it also has a type Mamdani output called fuzzy rate.
Graph-based user model	User modeling is not available	User modeling is not available	User modeling is available in this version based on graphs and is used to adapt inside diffuse inference aptitude function on ranking entrance.

5. **Combining the consequences to get an output distribution.**

6. **Defuzzifying the output distribution (this step is only if a crisp production (class) is needed).**

5.1.1 Interface.

An easy use individual evaluation web interface is develop to this version. We can notice in the navigation bar the access application functionality, worth mentioning that the app will only work with the social platform Facebook. Once accessed with a Facebook account the navigation bar will show an avatar as well as the exit application functionality through a logout. Also inside this interface exist the visual element Friends that will show all the participating friends in the form, to be able to view friends public collections is another functionality this element has. Continuing with the interface explanation, we found the Collections panel where the user's list of created and saved collections is shown as well as the functionality that give the ability to create new collections. An About this section will be shown inside; this will explain in a general way the functionality to the participating users. Now when it comes to users individual evaluation they will have a visual canvas element who will show behavior of a to be evaluated individual (animation) and in the lower part the individual will have the ability to star rank evaluate where one star means

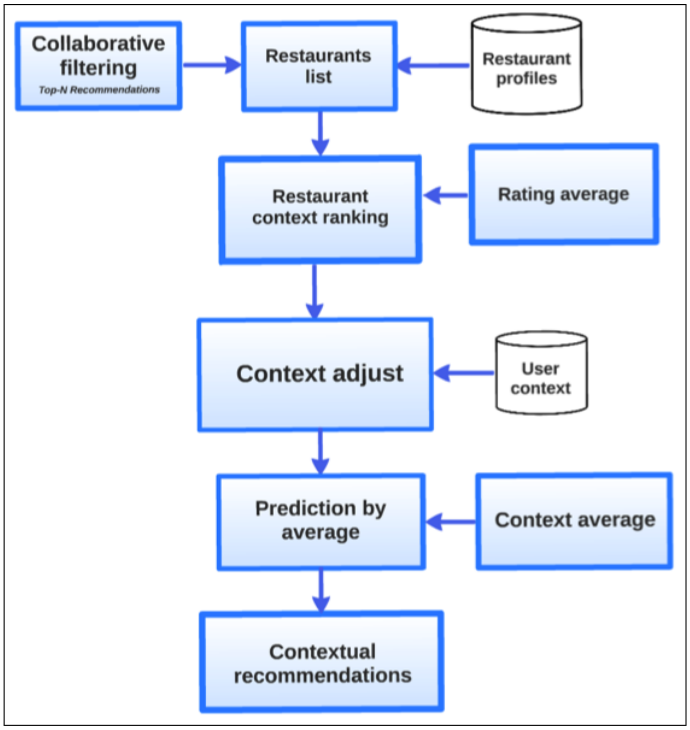

Figure 5.1: The Post-Filtering architecture for Tijuana restaurants.

he/she did not like it much and five stars meaning he/she like it too much. Now in the lower section of the star rank, a button will provide the ability to add that individual to a collection previously created. In the upper-rank star section, we will find a link which reads Click here to see my DNA History this will take us to detail about the individual to be evaluated that way we could review the detail of the individual in image x.

Detailed Important information about individual DNA history is shown here such as how many evaluations in likes he/she has received, how many visitors as well as ascendancy, genetic crossing operator, chromosome numeric representation

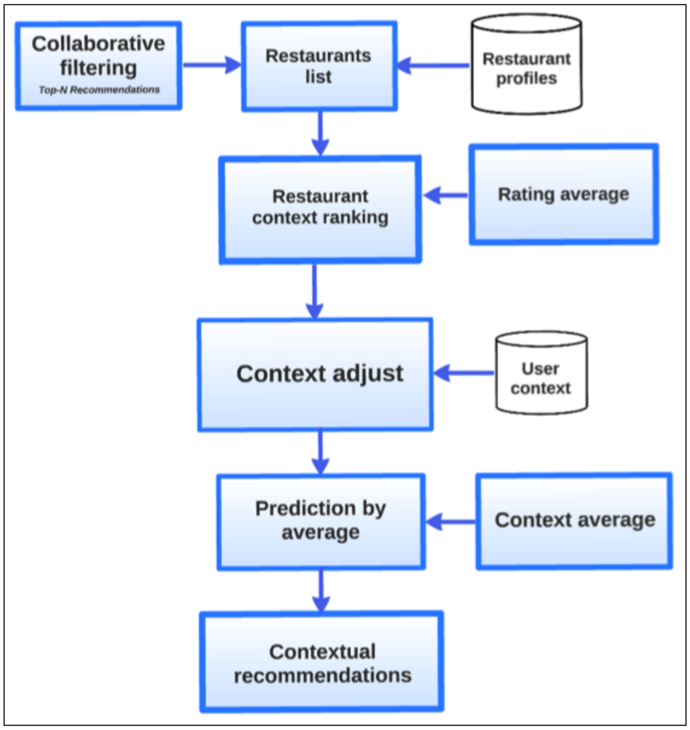

Figure 5.2: The Post-Filtering architecture for Tijuana restaurants.

and inside population identifier.

5.2 EvoDrawing02

Inside this version we will find an initial configuration of the interactive evolutionary algorithm in the following way:

As in the Las version it counts with 80 individuals as initial population and individuals are represented in the same way as in the previous version: Eight evaluations as one evolution parameter. One genetic Operator as aleatory selection between competition and ... One horizontal genetic crossing Operator. Its fitness function is provided by equation x.

1. **Having an initial 80 individual population which we will represent in the following way:**
2. **As in the Las version it counts with 80 individuals as initial population and individuals are represented in the same way as in the previous version.**
3. **Eight evaluations as one evolution parameter..**
4. **Combining the fuzzified inputs according to the fuzzy rules to establish a rule strength.**
5. **One genetic Operator as aleatory selection between competition and ...**
6. **Its fitness function is provided by equation x.**

5.2.1 Fuzzy Inferance.

Where represents all the individuals. Is the rank given to the individuals by the users. Represents one diffuse rank function composed by a diffuse inference system and will be represented by equation x.

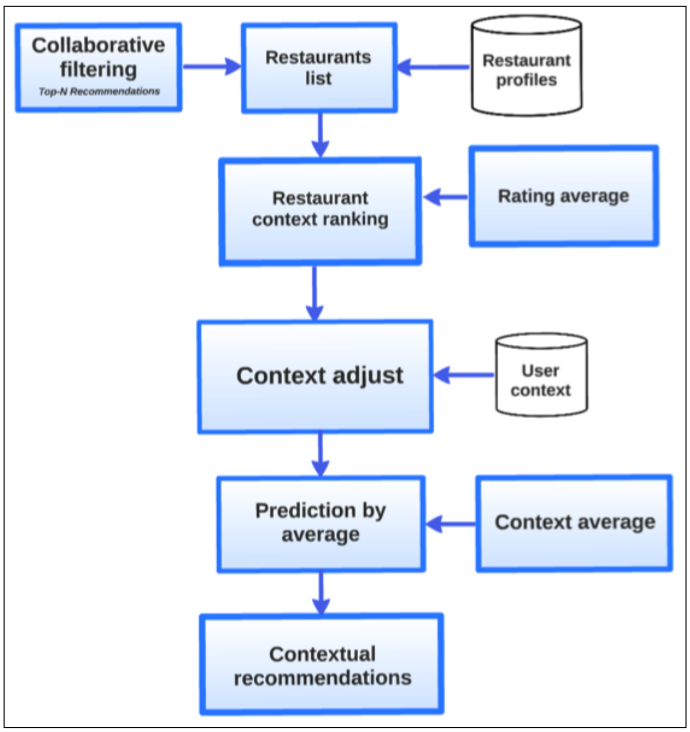

Figure 5.3: The Post-Filtering architecture for Tijuana restaurants.

Where represents the diffuse rank. Will continue been users given rank according to their preferences to each. Is the experience which is defined by the activity already explained in the previous chapter.

The diffuse inference system type is Mamdani, and it is composed by two entrances represented by and a fuzzy rate exit as we can observe represented in figure x.

It counts with 9 if/then rules which were designed in an empiric way and are the following:

If experience is low and preference is low then fuzzy_rate is bad If experience is mid and preference is low then fuzzy_rate is bad If experience is high and preference is low then fuzzy_rate is normal If experience is low and preference is mid then fuzzy_rate is bad If experience is mid and preference is mid then fuzzy_rate is normal If experience is high and preference is mid then fuzzy_rate is good If experience is low and preference is high then fuzzy_rate is normal If experience is mid and preference is high then fuzzy_rate is good If experience is high and preference is high then fuzzy_rate is good

5.2.2 Interface

The user interface in this version is the same as in the previous version. It only differs in how inference is made between versions.

5.3 EvoDrawing03

To create version we made an initial configuration in the following way:

1. **As in the previous version it counts with an 80 individuals initial population and individuals are represented in the exact same way as in the previous version.**
2. **8 evaluations as one evolution parameter.**
3. **One genetic Operator as aleatory selection between competition and**
4. **Combining the fuzzified inputs according to the fuzzy rules to establish a rule strength.**
5. **One horizontal genetic crossing Operator.**
6. **Its fitness function is provided by equation x.**

$$f = \frac{\sum_{i=0}^{n} x_i + f(y_i)}{\sum_{i=0}^{n} f(y_i)} \tag{5.1}$$

Where represents fuzzy rank. continues been the given rank users conceive to individuals according to their preferences. represents the experience defined by the activity already explained in the previous chapter. represents users ranking which we have previously described how is this variable calculated in the last chapter.

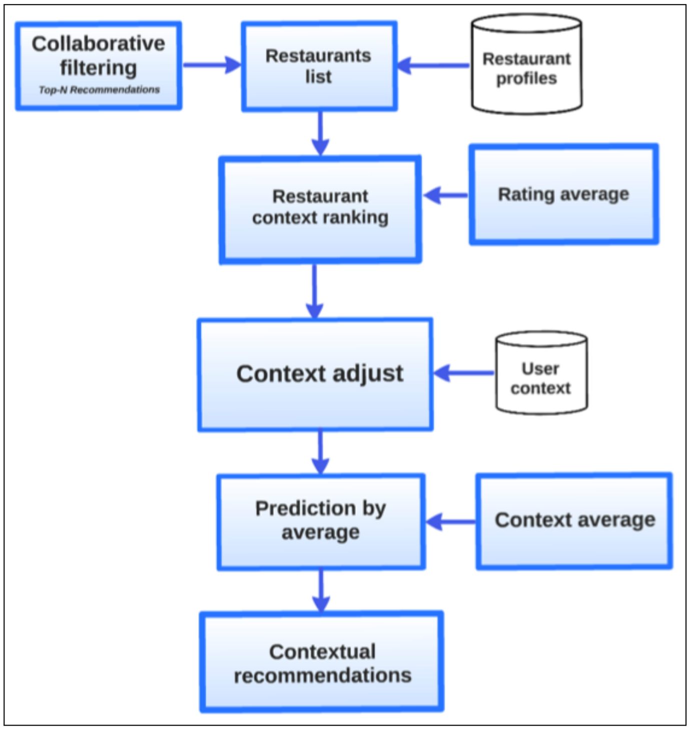

Figure 5.4: The Post-Filtering architecture for Tijuana restaurants.

The fuzzy inference system type is Mamdani and it is composed by three entrances represented by and a fuzzy rate exit as we can observe represented in figure x.

This version includes 30 diffuse rules designed in an empiric way. Two punctual differences exist in this version, the first one represented by the ranking value used in the users modeling based on graphs; the second one represents the usability elements adding a video game competitiveness paradigm. Inside figure X we will appreciate the interface design added to EvoDrawing 03 version.

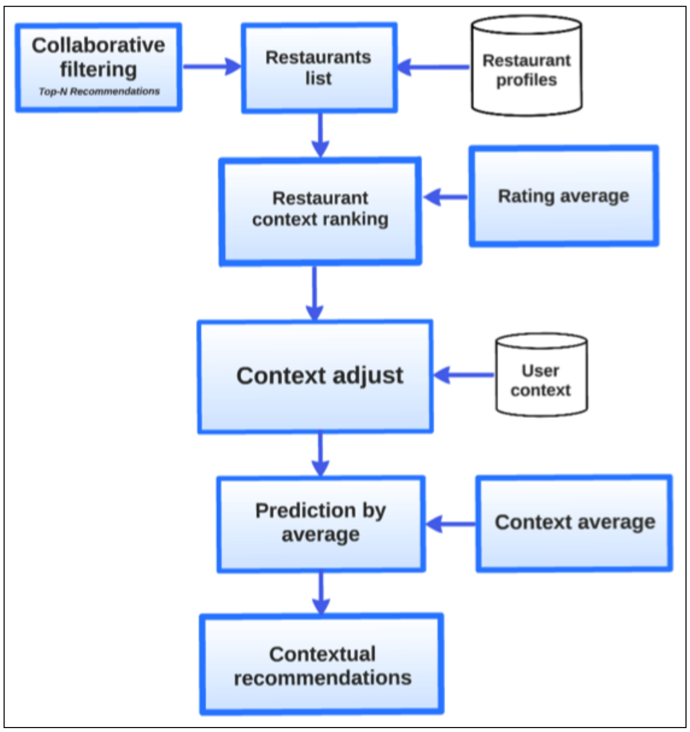

Figure 5.5: The Post-Filtering architecture for Tijuana restaurants.

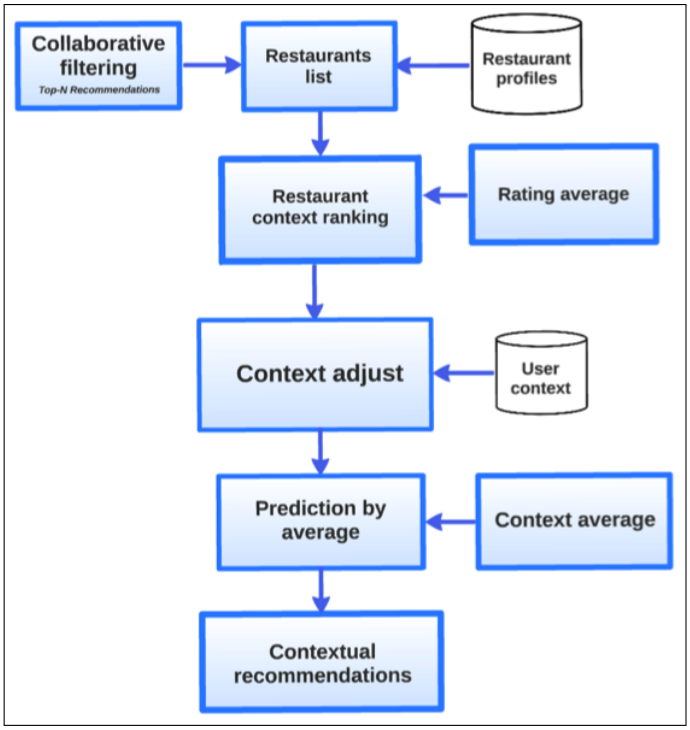

Figure 5.6: The Post-Filtering architecture for Tijuana restaurants.

Visual elements added to the navigation bar are added to this version, items such as the score where users can view its participations in EvoDrawings, it also provides you with the experience level acquired through their participation; This version will also offer an option to vie the leader chart inside the application as can we appreciate in figure x. The chart will show the ten best users and how many times they participate up to date.

Each experiment was promoted through social networking using specifically Facebook and Twitter por medio de un URL corto asi como tambien un anuncio para motivar a los usuarios a presionar el link (URL corto). Esto con el fin encontrar

Table 5.2: comparison of the different versions.

Long URL	Short URL
evodrawings01.herokuapp.com	goo.gl/J8TCe1
evodrawings02.herokuapp.com	goo.gl/jqjNy5
evodrawings03.herokuapp.com	goo.gl/J8TCe1

participantes de forma voluntaria. Este URL corto se puede observar en la tabla x.

Is Worth to mention each experiment were implemented through a cloud service by using the (Heroku) cloud service, in the following chart, each experiment will show its characteristics.

The experiments initial purpose is to design specific applications and interfaces to achieve user participation in a voluntary way in every single application. Each experiment allowed the commitment to obtain data to be used later to evaluate and analyze which version had the most user participation; To prove the initial hypothesis we implemented 3 study cases with slight differences in its configuration characteristics to each application allowing us enabling us to decide through obtained results which of the study cases prove this investigation hypothesis. Next chapter will show the Obtained results in the survey.

Table 5.3: characteristics of resources and services in cloud Heroku..

Service and resources	EvoDrawings01	EvoDrawings02	EvoDrawigs03
Heroku Free	✓	✓	✓
Deploy from Git	✓	✓	✓
Automated patching	✓	✓	✓
Self healing apps	✓	✓	✓
Undefined logs	✓	✓	✓
Number of process types	2	2	2
Always on Sleep after 30 mins of inactivity, otherwise always on depending on you remaining mostly free dynes hours	✓	✓	✓
Custom domains	✓	✓	✓
RAM 512	✓	✓	✓
Dedicated	x	x	x
Heroku Postgres ::DB Hobby Dev	✓	✓	✓
GrapheneDB Chalk	✓	x	✓
Redis To Go Nano	✓	x	✓
GrapheneDB Sandstone	x	x	✓
Redis To Go Mini	x	x	✓

Chapter 6

Results

In this chapter the results of the experiments described above are presented.

6.1 EvoDrawing01

The experiment EvoDrawing01 generated the following data shown in the table x

These data are generated total of nodes as well as relationships.

In order to observe which users have better social interconnectivity within the experiments it was decided to make a relation of number known users that the users

Table 6.1: Data generated in graph-based user modeling.

	Data
Nodes	595
Relationships	2220

Table 6.2: Total number of volunteers active users.

	Users
Total	53

Table 6.3: Number of known among users.

User name	Number of known users
Chriss de Blanc	7
Alejandro Salcido	4
Jonathan Amezcua Aguiluz	1
Daniela Sanchez	1
Jennifer Llamas	1
Iliana Dl	1
Xochilt Ramirez Garcia	1
Manuel Elizondo	1
Julian Torres	1
Data Back	1
Frank Arce	1
Gustavo Vargas	1

have. This relation is shown in table 3

Likewise in Table 3 interconnectivity having a particular user with other users is shown. The degree of relationship of users is associated with the number of friends known within the application. This means that we have the degree of influence among participants. For example the "Chriss Blanc" user has a degree of relatedness 7 (Figure x A)) indicating that this particular user can have more influence on the decisions of others. Moreover the user "Alejandro Salcido" has the second highest

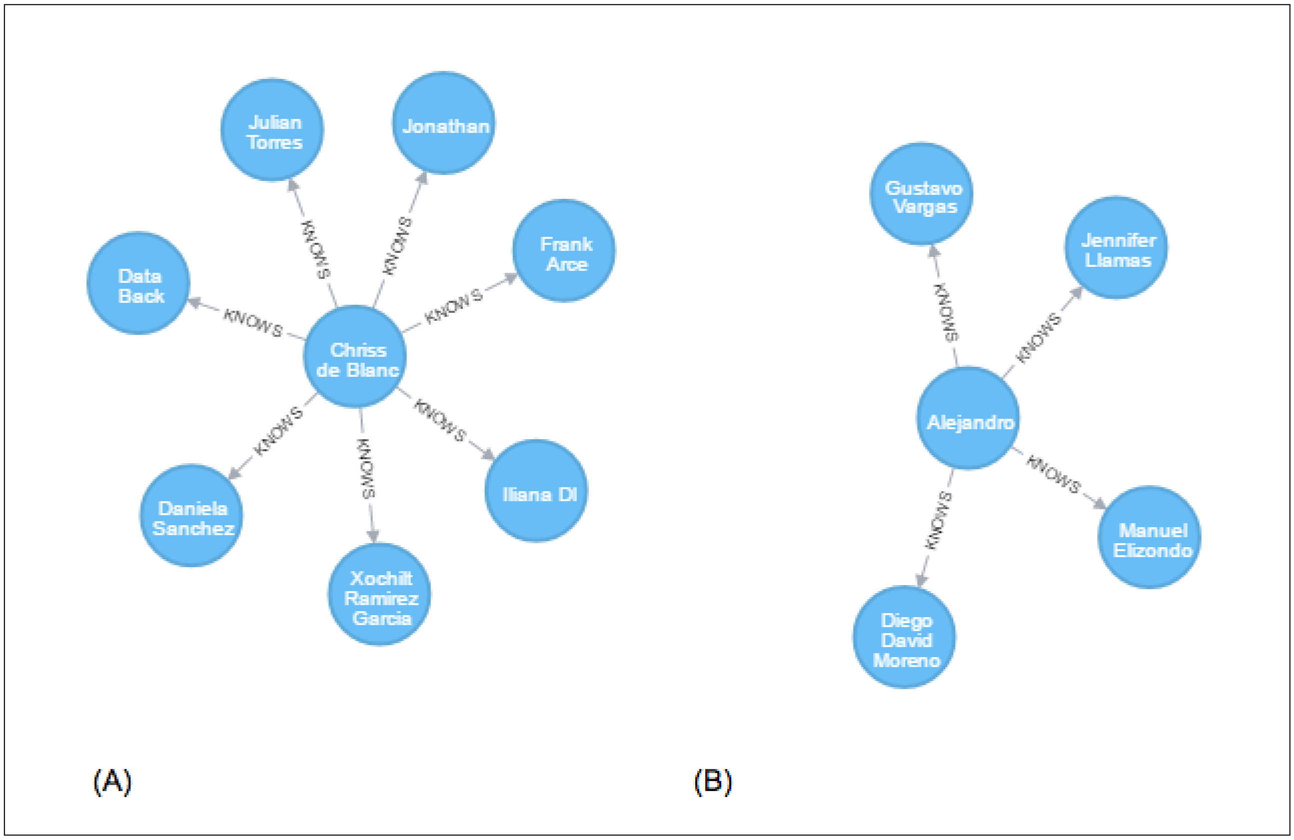

Figure 6.1: Graph users with greater interconnectivity.

degree of relationship to other users (Figure x B)).

In figure 6.1 shows users more connected in the graph. This may represent the degree of impact of a user and the possible influence that may have on the decisions of other users. For example, if the user with the greatest impact is affected in any decision likely other users may be affected in some way.

Table 4 presents the total number of individuals generated within the graph.

Table 5 contains a sample of 30/500 individuals were generated in the experiment.

Table 6.4: Total number of individuals generated.

	Individuals
Total	500

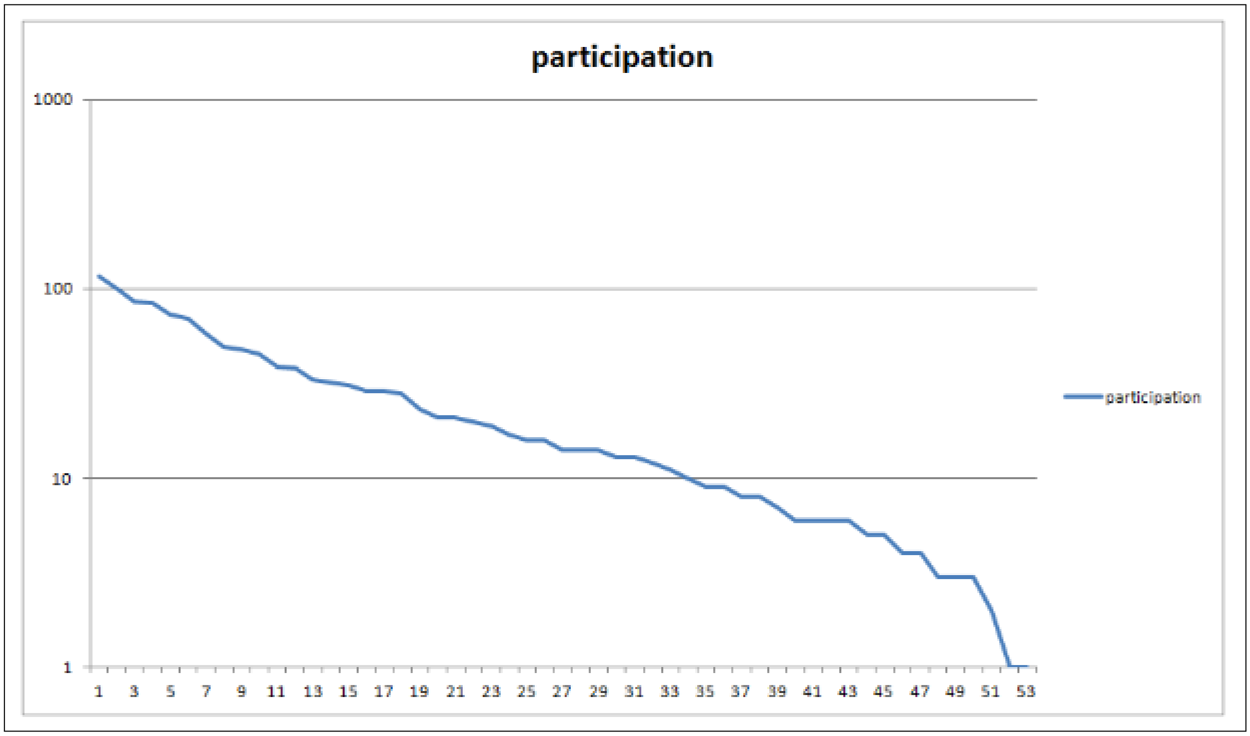

Figure 6.2: Visual representation of user participation in EvoDrawing01.

Where we present the unique identifier of the individual, its chromosome, as well as the number of views, likes available to the individual. This results are useful to observe individuals have been better evaluated by users.

Table 6 shows the results of the level of user participation in the experiment. These were obtained by counting the vicinity of nearest nodes from the base node in this case each user node.

Table 6.5: Sample of 30 individuals evaluated from.

id	Chromosome	Views	Likes
pop:individual:107	[125, 30, 0, 1, 0, 0, 3, 0, 1, 0, 0, 0, 0, 2, 1]	20	20
pop:individual:133	[143, 15, 1, 1, 0, 1, 3, 0, 1, 0, 2, 0, 0, 1, 1]	15	15
pop:individual:109	[125, 30, 1, 1, 0, 0, 3, 0, 1, 0, 0, 0, 0, 0, 1]	15	15
pop:individual:37	[87, 64, 1, 1, 1, 1, 4, 0, 1, 0, 0, 0, 1, 2, 3]	14	14
pop:individual:36	[95, 71, 0, 1, 1, 0, 3, 1, 0, 0, 0, 1, 1, 1, 2]	14	14
pop:individual:215	[138, 29, 1, 1, 0, 1, 3, 0, 1, 1, 1, 0, 0, 1, 1]	13	13
pop:individual:228	[42, 58, 0, 0, 1, 1, 4, 0, 0, 0, 2, 0, 0, 0, 3]	12	12
pop:individual:48	[51, 73, 0, 0, 0, 0, 2, 0, 0, 1, 3, 0, 1, 0, 3]	12	12
pop:individual:39	[60, 12, 1, 1, 1, 1, 4, 1, 1, 1, 0, 1, 0, 1, 1]	12	12
pop:individual:94	[49, 71, 0, 0, 1, 0, 4, 1, 0, 0, 1, 1, 1, 0, 2]	11	11
pop:individual:194	[87, 64, 0, 1, 1, 1, 1, 3, 0, 0, 0, 0, 1, 2, 3]	11	11
pop:individual:75	[97, 66, 0, 0, 1, 1, 3, 0, 1, 0, 3, 0, 1, 0, 1]	11	11
pop:individual:105	[125, 30, 0, 1, 0, 0, 3, 0, 1, 0, 0, 1, 0, 0, 1]	11	11
pop:individual:306	[87, 64, 1, 1, 1, 1, 4, 1, 1, 1, 1, 0, 1, 2, 3]	13	10
pop:individual:326	[138, 29, 1, 1, 0, 1, 3, 0, 1, 1, 1, 0, 0, 0, 0]	10	10
pop:individual:82	[81, 8, 1, 0, 0, 1, 4, 1, 1, 1, 2, 0, 0, 2, 1]	10	10
pop:individual:252	[125, 30, 0, 1, 0, 0, 3, 0, 1, 1, 3, 0, 1, 0, 3]	10	10
pop:individual:280	[53, 63, 1, 1, 0, 1, 3, 0, 1, 1, 0, 0, 0, 2, 1]	9	9
pop:individual:178	[53, 63, 1, 1, 0, 1, 3, 0, 1, 1, 0, 0, 0, 2, 1]	9	9
pop:individual:108	[42, 58, 0, 0, 1, 1, 3, 0, 1, 0, 1, 1, 0, 0, 1]	10	9
pop:individual:231	[125, 30, 0, 1, 0, 0, 3, 0, 1, 1, 3, 0, 1, 0, 3]	0	9
pop:individual:147	[122, 38, 1, 0, 0, 0, 0, 3, 0, 1, 0, 0, 0, 0, 0]	9	9
pop:individual:181	[143, 15, 1, 1, 0, 1, 3, 0, 1, 0, 0, 0, 0, 0, 1]	9	9

Table 6.6: Level of user participation.

User name	Participation
Ana Laura Lopez	116
Mario Garca Valdez	100
Chriss de Blanc	93
Xochilt Ramirez Garcia	85
Carlos David Gallardo Prez	73
Ulises Reus	70
Aaron Gutierrez Urbina	58
Cesar Lpez	49
Hector Beltran Medrano	48
Luis Alfonso Felix Garcia	45
Data Back	39
Amaury Hernandez Aguila	32
Osmar Herrera Duran	31
Jorman Gtz	29
Alexis Campos Lopez	29
Melissa Muoz Montes	28
David Gallegos	23
Jose Carlos	21
Toms Perrn	21
Manuel Elizondo	20

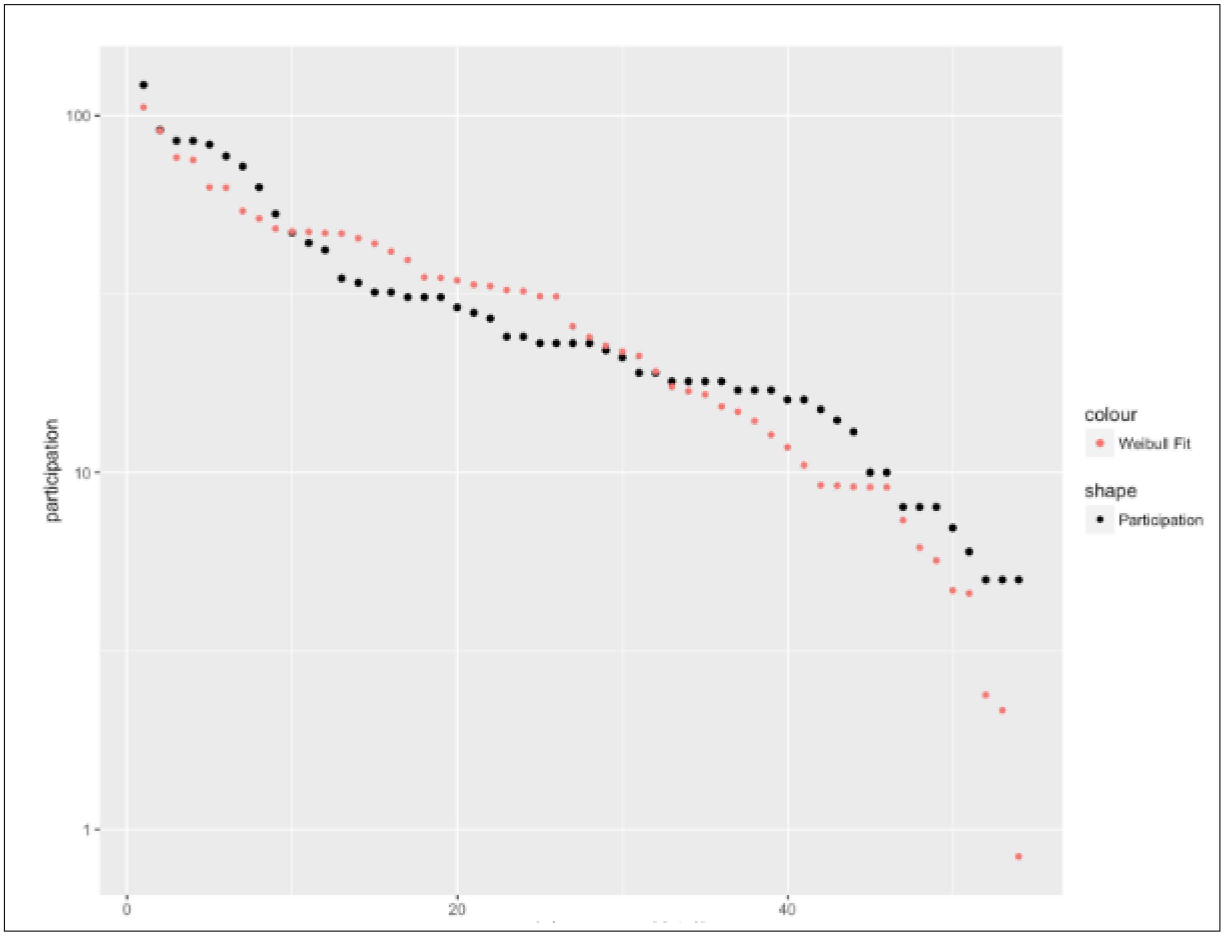

Figure 6.3: Weibull fit data representation.

In figure 6.2 we present a visual representation of user participation this experiment where the y-axis represents the level of participation and the x axis represents the number of users who participated in this experiment.

Table 6.7: Data generated in graph-based user modeling.

	Data
Nodes	648
Relationships	2596

Table 6.8: Total number of volunteers active users.

	Users
Total	54

6.2 EvoDrawing02

In the experiment EvoDrawings02 the following data were generated presented in Table x.

These data are generated total of nodes as well as relationships.

The total number of users who participate voluntarily shown in the table x.

Like the previous experiment which wanted to observe users had better social interconnectivity within this experiment, it was decided to make a relationship of the number of known users. This relationship presented in table x where the top 10 user with social interconnectivity are shown.

In figure 3 shows users more connected in the graph. This may represent the degree of impact of a user and the possible influence that may have on the decisions of other users. For example, if the user with the greatest impact is affected in any decision likely other users may be affected in some way. Particularly in this

Table 6.9: A sample of the top 10 users level influence on users.

User name	Number of known users
Rogelio UR	10
Barbara Sandoval	9
Evelyn Macedo	8
Chriss de Blanc	8
Cesar Rojas	8
Jasiel Calzada	8
Tonyy Maldonado	7
Silvano Peraza	7
Hector Beltran Medrano	6
Juan Ferman Lopez	6

experiment the degree of social interconnectivity among users is becoming more complex as compared to the previous experiment.

Table 9 contains a sample of individuals 30/556 generated in the experiment.This table shows the unique identifier of the individual, its chromosome, as well as the number of views, likes available to the individual.This results are useful to observe individuals have been better evaluated by users.

In table 10 shows the results of the level of user participation in the experiment.These were obtained by counting the vicinity of nearest nodes from the base node in this case each user node.

In Figure 3 we present a visual representation of user participation of this experiment where the y-axis represents the level of participation and the x axis represents

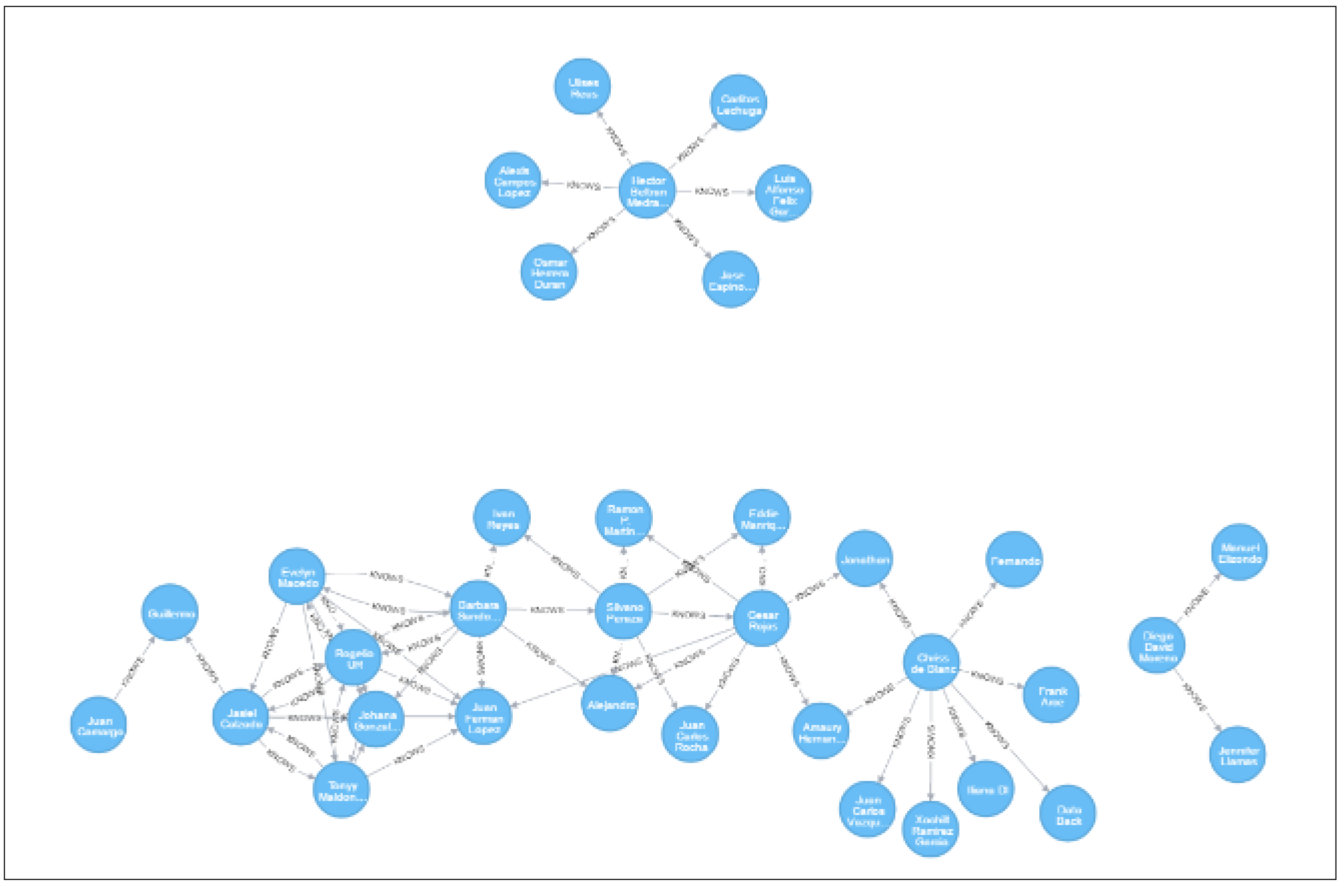

Figure 6.4: Sample of the top 30 individuals evaluated from a total of 556.

Table 6.10: Sample of 30 individuals evaluated from.

id	Chromosome	Views	Likes
pop:individual:55	[63, 58, 0, 1, 1, 1, 4, 0, 0, 1, 0, 0, 0, 0, 3]	18	18
pop:individual:329	[98, 37, 0, 1, 1, 0, 4, 0, 0, 1, 0, 0, 0, 0, 2]	17	17
pop:individual:304	[63, 58, 0, 1, 1, 1, 4, 0, 0, 0, 3, 0, 0, 2, 2]	16	16
pop:individual:202	[107, 79, 1, 0, 1, 1, 3, 0, 0, 1, 3, 0, 1, 1]	15	15
pop:individual:58	[150, 79, 1, 0, 1, 1, 3, 0, 0, 1, 3, 0, 1, 1, 1]	14	14
pop:individual:310	[63, 58, 0, 1, 1, 1, 4, 0, 0, 1, 0, 0, 1, 2, 2]	13	13
pop:individual:67	[65, 51, 1, 1, 0, 0, 3, 1, 0, 0, 3, 1, 0, 2, 3]	12	12
pop:individual:48	[51, 73, 0, 0, 0, 0, 2, 0, 0, 1, 3, 0, 1, 0, 3]	12	12
pop:individual:179	[98, 37, 0, 1, 1, 1, 4, 0, 0, 1, 0, 0, 0, 0, 3]	12	12
pop:individual:114	[63, 58, 0, 1, 1, 1, 4, 0, 0, 1, 0, 0, 0, 0, 2]	12	12
pop:individual:123	[124, 42, 0, 1, 1, 1, 4, 1, 0, 1, 1, 0, 0, 2, 1]	12	12
pop:individual:344	[97, 66, 0, 0, 1, 1, 3, 0, 1, 0, 3, 0, 1, 0, 1]	11	11
pop:individual:105	[63, 58, 0, 1, 1, 1, 4, 0, 0, 1, 0, 0, 0, 0, 2]	11	11
pop:individual:435	[98, 37, 0, 1, 1, 0, 1, 1, 4, 1, 0, 0, 0, 0, 2]	11	11
pop:individual:140	[150, 79, 1, 0, 1, 1, 3, 1, 1, 0, 1, 1, 1, 2, 3]	11	11
pop:individual:216	[124, 42, 0, 1, 0, 0, 3, 0, 0, 1, 1, 0, 0, 1, 1]	11	11
pop:individual:290	[150, 79, 1, 0, 1, 1, 3, 0, 0, 1, 3, 0, 1, 0, 1]	10	10
pop:individual:255	[150, 79, 1, 0, 1, 1, 3, 1, 1, 1, 0, 0, 1, 2, 2]	10	10
pop:individual:14	[89, 66, 0, 0, 1, 1, 3, 0, 0, 0, 2, 0, 0, 2, 1]	10	10
pop:individual:366	[128, 66, 0, 1, 1, 4, 1, 0, 1, 0, 0, 0, 1, 2, 2]	10	10
pop:individual:215	[54, 72, 0, 1, 1, 1, 4, 0, 0, 1, 0, 0, 0, 0, 2]	9	9
pop:individual:411	[113, 58, 0, 1, 1, 1, 4, 0, 0, 1, 0, 0, 1, 2, 2]	9	9
pop:individual:254	[113, 58, 0, 1, 1, 4, 1, 0, 4, 1, 1, 0, 0, 3]	11	9

Table 6.11: Level of user participation.

User name	Participation
1122212314475816	122
1107674982600275	91
10207544086753085	86
1128990193799346	85
1067084180030552	83
985591718197586	77
969507913124553	72
10207004677610003	63
1223229694371825	53
10153904046011462	47
1032494570125948	44
995610090523549	42
975038365907627	35
10207487295119454	34
1041989022528624	32
471974623003503	32
1275844349097672	31
978744228875903	31
10205734318020434	31
10209454397419860	29

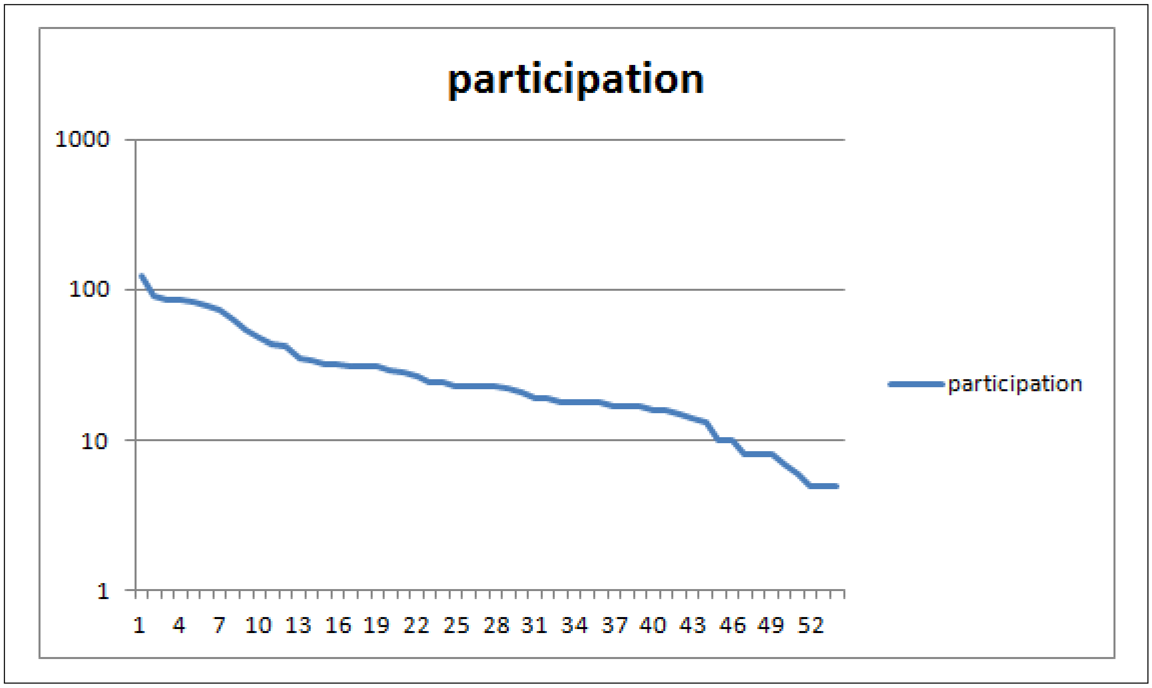

Figure 6.5: Visual representation of user participation in EvoDrawing01.

Table 6.12: Data generated in graph-based user modeling.

	Data
Nodes	3746
Relationships	17207

the number of users who participated in this experiment.

6.3 EvoDrawing03

In the experiment EvoDrawings03 the following data were generated presented in Table x.

The total number of users who participate voluntarily shown in table x.

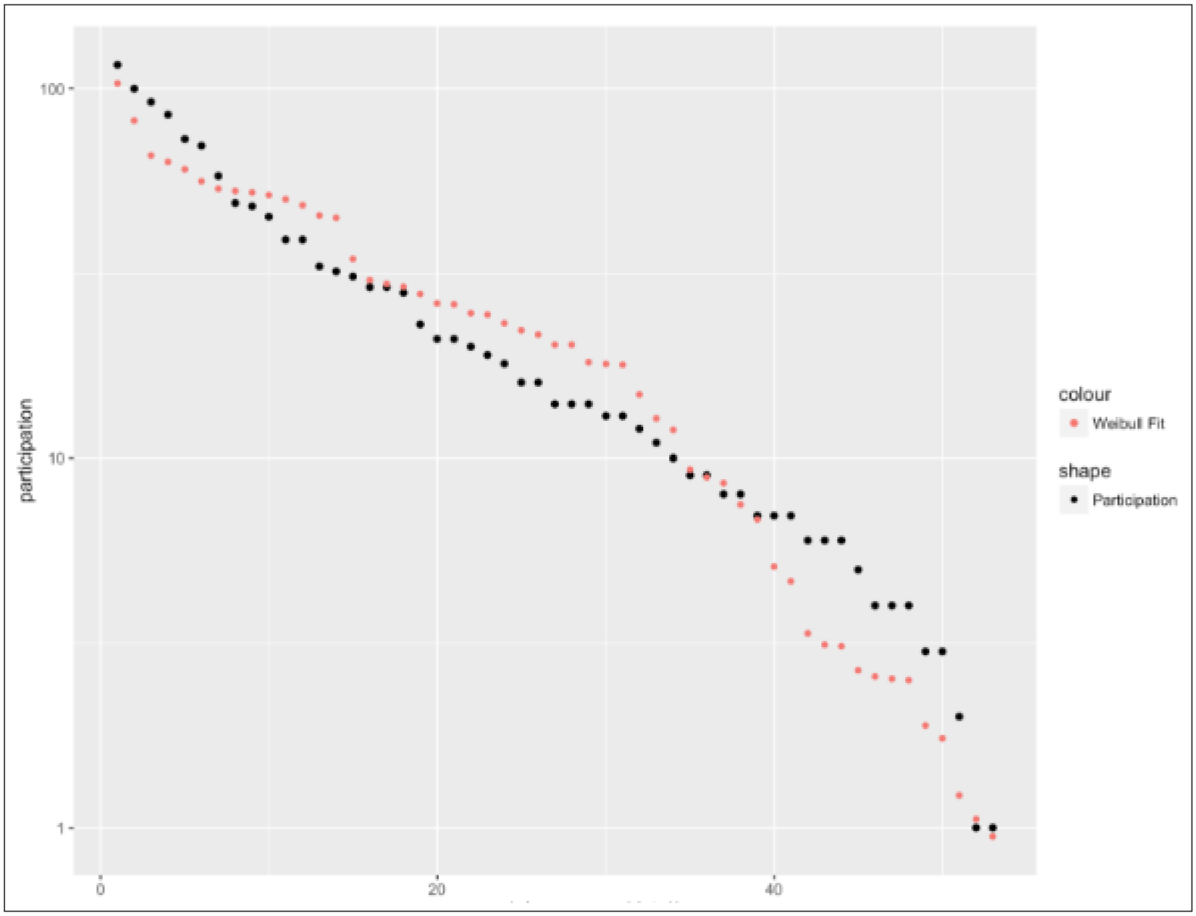

Figure 6.6: Weibull fit data representation.

Table 6.13: Total number of volunteers active users.

	Users
Total	68

Table 6.14: A sample of the top 10 users level influence on users.

User name	Number of known users
David Astorga Viramontes	23
Evelyn Macedo	18
Reyes Zuniga	17
Barbara Sandoval	16
Chriss de Blanc	15
Luis Alfonso Felix Garcia	15
Paulina NG	15
Marco Antonio Fuentes Alvarez	15
Jasiel Calzada	14
Hector Beltran Medrano	13

In the same way as the previous experiment which wanted to observe users had better social interconnectivity within this experiment, it was decided a known ratio of the number of users that have. This relationship is presented in Table 6.14 13 where we show the 10 users with better social interconnectivity.

In figure 6.7 shows users more connected in the graph. This may represent the degree of impact of a user and the possible influence that may have on the decisions of other users. For example, if the user with the greatest impact is affected in any decision likely other users may be affected in some way. Particularly in this experiment the social graph interconnectivity among users is becoming more complex as compared to the previous experiment as we can observe that more closely

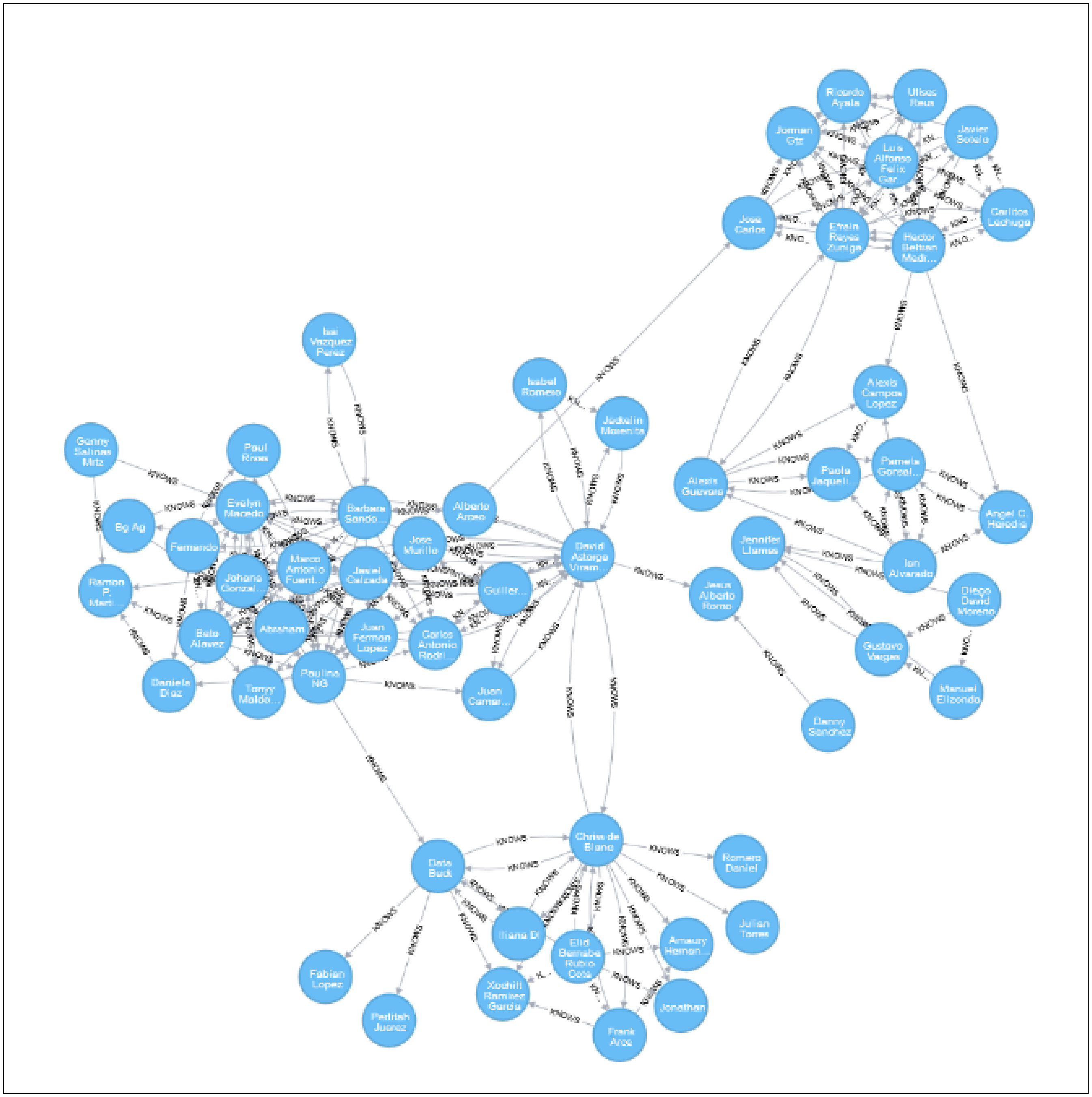

Figure 6.7: Social interconnectivity in graph-based user meodel.

Table 6.15: Total number of individuals.

	Individuals
Total	3594

resembles a social network.

In table 6.18 shows the total number of individuals evaluated by users in this experiment.

Table 6.18 contains a sample of individuals 30/556 generated in the experiment. In this its unique identifier of the individual, its chromosome, as well as the number of views, likes available to the individual presents. This results which are useful to observe individuals have been better evaluated by users.

In Figure 3 we present a visual representation of user participation of this experiment where the y-axis represents the level of participation and the x axis represents the number of users who participated in this experiment.

6.4 Comparison between experiments.

In chart 6.10 shows a comparative graph of the results obtained in the three phases of the study case, each of the lines on the graph represents one of the versions of EvoDrawing in relation to the number of units of users that were obtained are presented in different experiments. For instance the blue line represents EvoDrawing01

Table 6.16: Sample of 30 individuals evaluated from.

id	Chromosome	Views	Likes
pop:individual:3570	[113, 44, 0, 1, 0, 0, 1, 0, 1, 0, 3, 1, 1, 0, 1]	48	48
pop:individual:3544	[150, 44, 0, 0, 1, 1, 1, 0, 0, 1, 1, 0, 1, 0, 0]	32	26
pop:individual:210	[63, 58, 0, 1, 1, 1, 4, 0, 0, 0, 3, 0, 0, 2, 2]	16	16
pop:individual:202	[113, 5, 0, 1, 1, 0, 4, 0, 1, 1, 2, 1, 0, 2, 2]	22	22
pop:individual:58	[61, 70, 0, 0, 1, 0, 1, 0, 0, 0, 2, 0, 1, 0, 3]	23	21
pop:individual:858	[113, 1, 1, 1, 1, 0, 1, 1, 1, 3, 0, 1, 1, 0]	21	21
pop:individual:17	[113, 69, 0, 1, 1, 1, 3, 0, 1, 1, 0, 0, 1, 0, 3]	23	21
pop:individual:3636	[118, 44, 0, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 0]	22	21
pop:individual:33	[129, 79, 0, 1, 1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 1]	20	20
pop:individual:74	[150, 50, 1, 1, 1, 1, 1, 1, 0, 1, 3, 1, 0, 1, 3]	21	19
pop:individual:65	[94, 62, 1, 0, 0, 0, 2, 0, 1, 0, 3, 0, 1, 2, 2]	20	19
pop:individual:50	[50, 70, 0, 1, 1, 1, 3, 1, 1, 1, 1, 1, 1, 1, 3]	19	19
pop:individual:52	[131, 63, 1, 1, 1, 0, 4, 0, 1, 1, 0, 0, 1, 0, 3]	20	19
pop:individual:2449	[113, 44, 0, 1, 1, 1, 1, 0, 1, 1, 0, 0, 1, 1, 2]	18	18
pop:individual:44	[117, 54, 0, 0, 0, 0, 2, 0, 1, 1, 3, 1, 0, 2, 3]	19	18
pop:individual:113	[77, 21, 1, 0, 1, 0, 4, 0, 1, 0, 0, 1, 0, 3]	17	17
pop:individual:84	[102, 69, 1, 0, 1, 0, 3, 0, 0, 0, 0, 0, 1, 0, 3]	17	17
pop:individual:95	[68, 43, 0, 1, 0, 1, 3, 1, 0, 0, 3, 0, 0, 1, 2]	19	16
pop:individual:22	[137, 20, 0, 0, 0, 0, 0, 1, 0, 0, 3, 0, 1, 1, 1]	17	16
pop:individual:71	[79, 5, 0, 0, 1, 1, 2, 1, 0, 1, 0, 1, 1, 1, 3]	16	16
pop:individual:42	[112, 35, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1, 1, 0, 2]	9	9
pop:individual:411	[113, 58, 0, 1, 1, 1, 4, 0, 0, 1, 0, 0, 1, 2, 2]	16	15
pop:individual:59	[93, 23, 0, 0, 0, 1, 4, 0, 0, 0, 1, 0, 1, 2, 1]	17	15

Table 6.17: Level of user participation.

User name	Participation
1157401414272355	2035
1001585659925992	1828
1244770712203948	1722
990632971026794	552
987920194610578	456
10207675081608741	300
966757780068615	267
1228561717171956	258
985416538190778	203
1123938694291507	202
10207552420841432	200
1124138344283213	169
220415891643957	161
534039336778842	112
10153940958696462	93
10205543461172072	87
1281817738500333	81
10153866481615259	74
943775989063530	73
10205664691039169	71

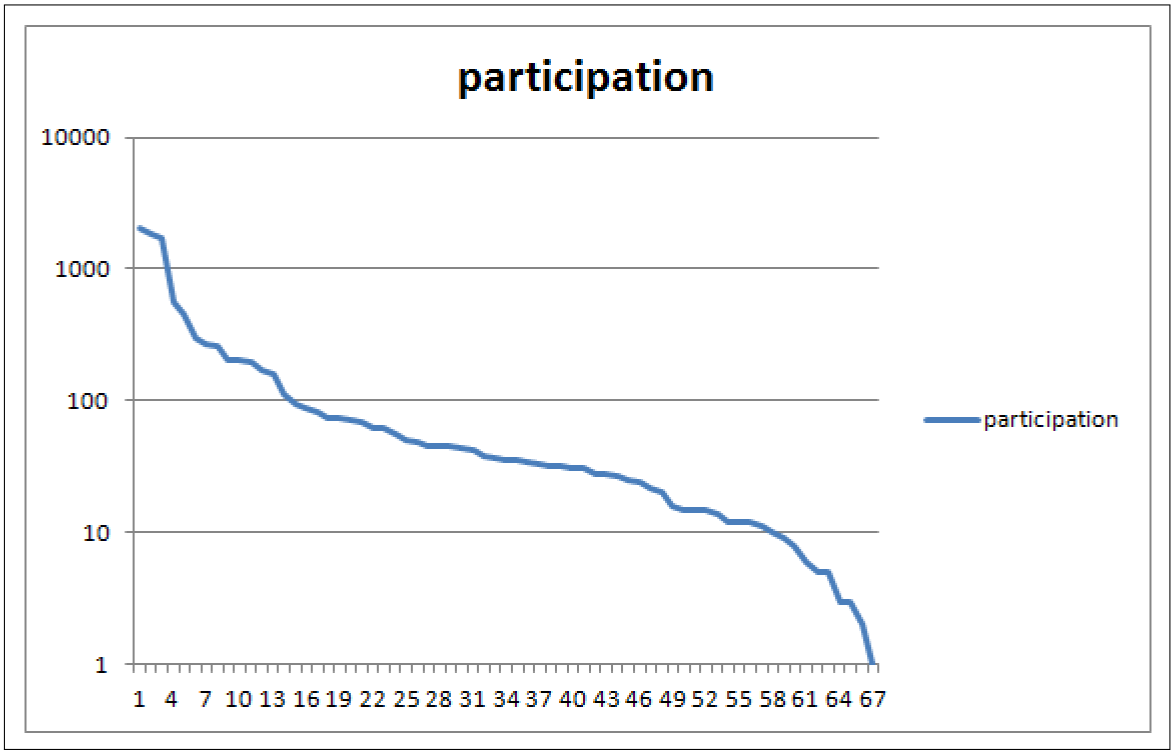

Figure 6.8: Visual representation of user participation in EvoDrawing01.

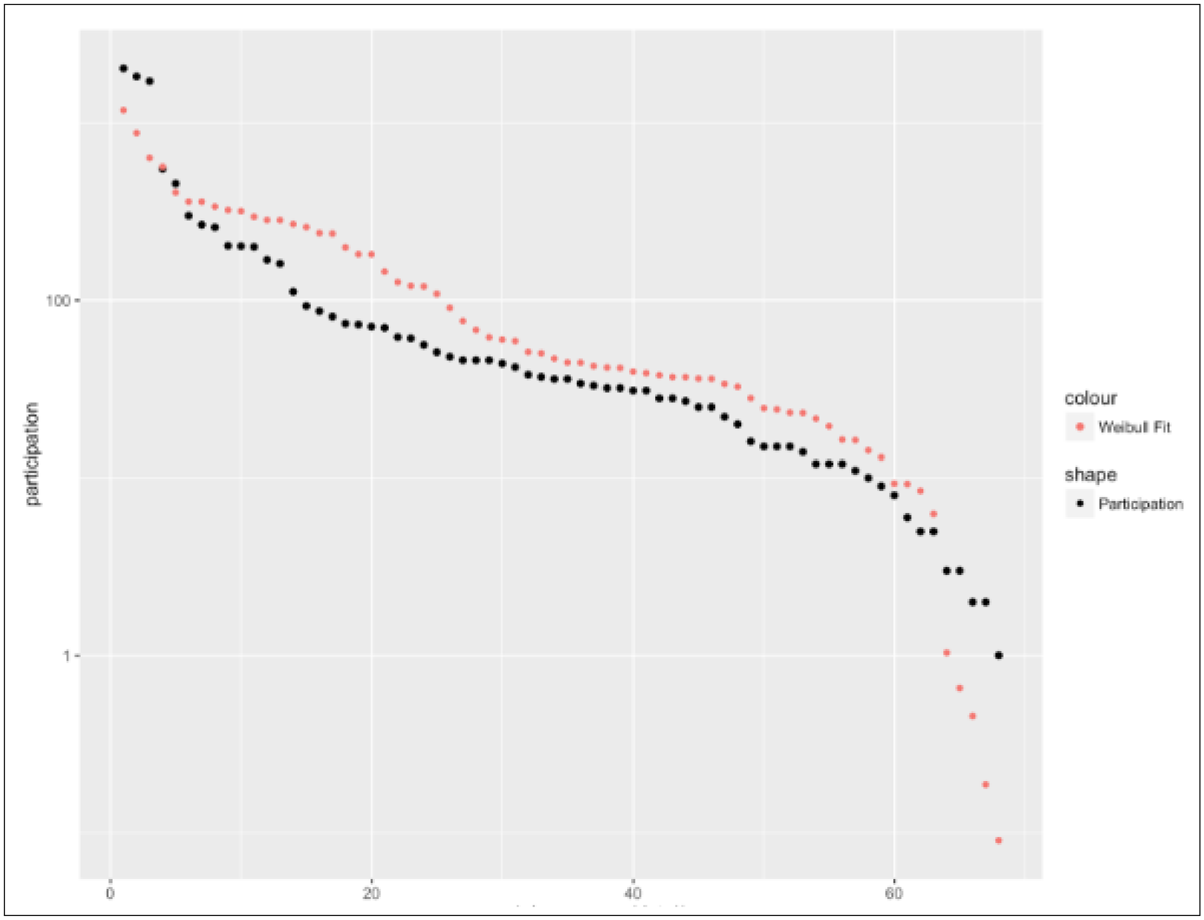

Figure 6.9: Weibull fit data representation.

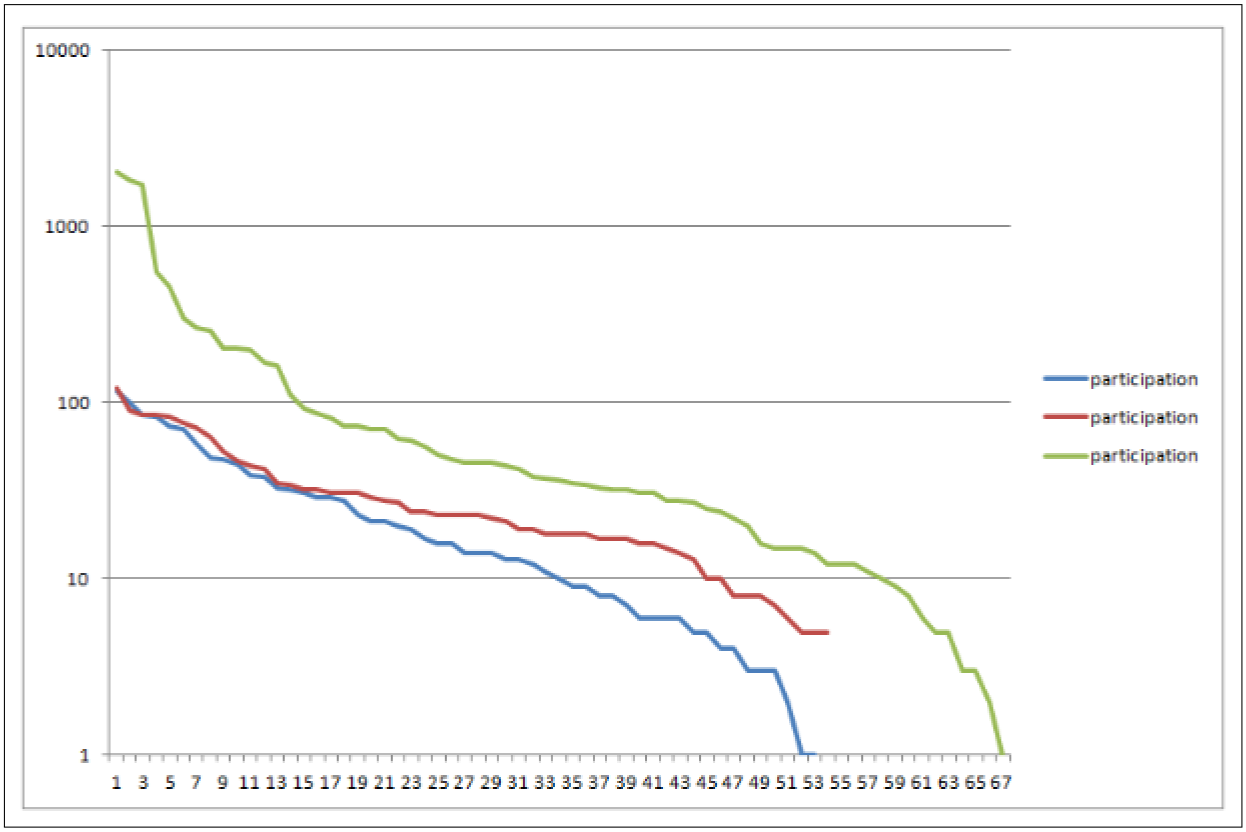

Figure 6.10: Graphical representation of user participation in the different experiments.

experiment, the red line represents EvoDrawing02 and finally the green line represets EvoDrawing03.

Table x the results of each experiment are described from column 1 to column 3 represents the number of shares of each user in each experiment respectively. The last three columns represent the difference of participations between experiments. In the figure and table x x an important phenomenon is observed, the difference between the participations of users in the first and second experiment is not wide, even in some users perceive that the first experiment got more shares.The important observable phenomenon is the difference between the shares of the top users of third experiment with respect to the first and second experiment, it is a very noticeable difference. The difference is attributed to the implementation of a model of ”Gamification” adapted at the interface of the third experiment. It is presumed that this model encourages users to participate more in the interfaces due to the competitive nature among users.

Table 6.18: Difference between the top 30 user participations of the different experiments.

EvoDrawings01	EvoDrawings02	EvoDrawings03	Difference between ED02 and ED01	Difference between ED03 and ED01	Difference between ED03 and ED02
116	122	2035	6	1919	1913
100	91	1828	-9	1728	1737
93	86	1722	-7	1629	1636
85	85	552	0	467	467
73	83	267	10	383	373
70	77	258	7	230	223
58	72	203	14	209	195
49	63	202	14	209	195
48	53	200	5	155	150
45	47	169	3	157	155
39	44	161	2	161	156
39	42	112	2	130	127
33	35	93	1	128	126
32	34	87	3	80	78
31	32	81	2	62	61
29	31	74	3	58	55
29	31	73	8	52	50
28	31	71	8	46	43
23	29	70	7	50	42
21	28	62	7	50	42
21	28	70	7	49	42
20	27	62	7	42	35
19	24	61	5	42	37
18	24	56	6	38	32
16	23	51	7	35	28
16	23	48	7	32	25
14	23	46	9	32	23
14	22	46	9	32	23
14	22	46	8	32	24
13	21	44	8	31	23

Chapter 7

Conclusions and future work

7.1 Conclusions

Using a user model in Web-based interactive evolutionary computation overall with the different approaches such as fuzzy logic and gamification it demonstrated in experiment EvoDrawing03 that the users increase their participation with respect to other versions (EvoDrawing01, EvoDrawin02).

In this sense the results have shown a phenomenon in the users which is competitiveness. This phenomenon occurs naturally because as human beings is our nature to be competitive regardless of the topic or activity that we assign [Reference]. This gave support to users return to evaluate more individuals within the experiment EvoDrawings03 and consequently the participation increase exponentially.

In this research work also found that the way individuals was presented to be evaluated and how to evaluate them helped the user to take their evaluations so easy and quick. This means that users evaluated on average 40 or more individuals in an iteration, reducing the risk of demotivation assessment and therefore lose interest in participation. However, it was concluded that the biggest problem of interactive evolutionary computing systems remains on user fatigue.

The fatigue can be generated by many factors, such as how to evaluate individuals, the subject of the application, how to present the individuals, the objective within the application, expertise by users, and more. The resulting method of this research helps motivate users on the issue of participation in interactive evolutionary computing applications.

7.2 Future work

Competitiveness is among users regardless of the level of expertise they have. Still psychological test would be necessary to see the level of fatigue that users acquire in systems of this context. These psychological tests are beyond the scope of this thesis since the main objective of the research was about measuring the participation of users through a graph-based user modeling, which finally answers the hypothesis of this research. However this research allows adaptation and implementation of other techniques and research that enrich this work with multidisciplinary teams to determine that this method and other methods can reduce the user's fatigue in the context of Web-based interactive evolutionary systems.

Another way to attract and increase users participations is to work with techniques of intelligent interfaces and natural interfaces in order to the user's evaluate more naturally, thus creating more natural and intelligent way to evaluate individuals, with which the user may feel more comfortable in evaluating individuals. For instance gesture-based interfaces, visual interfaces using sensors (cameras, Kinnect) to detect the time or emotion felt by the user when presented with an individual who needs to be evaluated.

The combination of these techniques generates more robust methods in competitiveness among users using this type of interfaces, for example implement rewards medals type or representative plates when the user reaches a certain level of participation it increases the interest to continue participating and improving the level of expertise within the systems, as well as to share their achievements in their social networks. In this sense we believed that no matter the topic of interactive evolutionary computation systems users will participate collaboratively having fun without knowing in depth is participating.

Publications

1. *Restaurant Recommendations based on a Domain Model and Fuzzy Rules. Xochilt Ramírez-García, Mario García-Valdéz. International Seminar on Computational Intelligence. Tijuana Institute of Technology. Tijuana Mexico. (2012).*

2. *Post-filtering for a Context-Aware Recommender System. Xochilt Ramírez-García, Mario García-Valdéz. Recent Advances on Hybrid Approaches for Designing Intelligent Systems . Springer International Publishing Switzerland. (2013).*

3. *Recomendaciones contextuales basadas en el enfoque de post-filtrado. Xochilt Ramírez-García, Mario García-Valdéz. Modelado computacional de Habilidades Linguisticas y Visuales. Vol.74. Research in Computer Sciences, IPN. 2014.*

4. *Context-aware Recommender System Based in Pre-filtering Approach and Fuzzy*

Rules. Xochilt Ramírez-García, Mario García-Valdéz. Recent Advances on Hybrid Approaches for Designing Intelligent Systems . Springer International Publishing Switzerland. (2014).

5. *Context-Aware Recommender System Using Collaborative Filtering, Content-Based Algorithm and Fuzzy Rules. Xochilt Ramírez-García, Mario García-Valdéz, 2016.*

6. *A Hybrid Context-aware Recommender System for Restaurants. Xochilt Ramírez-García, Mario García-Valdéz, 2016.*

Appendix A

Pseudocode

Algorithm 1 Get Cosine similarity values

Require: The list of itemProfilesUser and itemProfilesAll in binary format.
Ensure: The list of cosine similairty value for each item of the itemProfilesUser with each element of itemProfilesAll.

$allProfiles \leftarrow$[ ]
for $itemu$ to size of $itemProfilesUser$ **do**
  for $itema$ to size of $itemProfilesAll$ **do**
    if $itemu = itema$ **then**
      jump next item
    else
      $cosineSimilarityValue \leftarrow$ among $itemu$ and $itema$
      $itemProfiles \leftarrow itemu, itema, cosineSimilarityValue$
    end if
  end for
end for
return $allProfiles$

Algorithm 2 Collaborative filtering algorithm

Require: The userId.
Ensure: The Top-N list of recommendations for the current user.
  $ratingMatrix \leftarrow allRatings$
  Call *Recommendations* $\leftarrow$ *getRecommendations*() module
  return *Recommendations*

Algorithm 3 Content-Based Algorithm

Require: The user id.
Ensure: The Top-N list of recommendations.
  RV $\leftarrow$ All items that user rated with 5
  for *item* to size of *RV* **do**
    if *item* is not in *RV* **then**
      UV $\leftarrow$ *itemid*
    end if
  end for
  allItems $\leftarrow$ [ ]
  $getItemsProfilesUser \leftarrow$ Binary vectors of RV
  allRatings $\leftarrow$ Rating matrix
  for *item* to size of *allRatings* **do**
    if *itemid* is not in *allItems* **then**
      allItems $\leftarrow$ *item*
    end if
  end for
  $getAllItemsProfiles \leftarrow$ Binary vectors of allItems
  getCosineSim $\leftarrow$ getItemsProfilesUser,getAllItemsProfiles
  for *item* to size of *highCosineSim* **do**
    if *itemsimilarity* ≥ 0.8 **then**
      highCosineSim $\leftarrow$ *item*
    end if
  end for
  Sort *highCosineSim* list
  return $itemProfiles$

Algorithm 4 Get item profiles

Require: The UV vector, allItems vector and boolean value of userProfile.
Ensure: The list of temProfiles in binary vectors.

```
if userProfile true then
    getItemsProfilesUser ← UV
    for itemp to size of UV do
        get binary vector of itemp
        itemProfiles ← itemp
    end for
else
    allItemProfiles ← allItems
    for itemp to size of allItems do
        get binary vector of itemp
        itemProfiles ← itemp
    end for
end if
return itemProfiles
```

Algorithm 5 Calculate Cosine similarity

Require: The itemProfileUser and itemProfileAll, both vectors in binary format.
Ensure: The cosine similarity value.

```
sum ← 0
normaItemUser ← 0
normaItemAll ← 0
for position to size of itemProfileUser do
    sumProduct ← sumProduct + (itemProfileUser[position] * itemProfileAll[position])
end for
for item to size of itemProfileUser do
    normaItemUser ← normaItemUser + itemProfileUser[item]^2
end for
for item to size of itemProfileAll do
    normaItemAll ← normaItemAll + itemProfileAll[item]^2
end for
squareRootUser ← squareroot(normaItemUser)
squareRootAll ← squareroot(normaItemAll)
cosineSimilarity ← sumProduct/(squareRootUser * squareRootAll)
return cosineSimilarity
```

Algorithm 6 Create a binary vector of item profile

Require: The tem profile content in r.
Ensure: The temProfile of r in a binary vector.

$price \leftarrow [4]$
$payment \leftarrow [2]$
$alcohol \leftarrow [2]$
$smokingarea \leftarrow [2]$
$dresscode \leftarrow [3]$
$parking \leftarrow [3]$
$installation \leftarrow [4]$
$atmosphere \leftarrow [5]$
$cuisine \leftarrow [30]$
$price[positionPriceId - 1] \leftarrow 1$
$payment[positionPriceId - 1] \leftarrow 1$
$alcohol[positionPriceId - 1] \leftarrow 1$
$smokingarea[positionPriceId - 1] \leftarrow 1$
$dresscode[positionPriceId - 1] \leftarrow 1$
$parking[positionPriceId - 1] \leftarrow 1$
$installation[positionPriceId - 1] \leftarrow 1$
$atmosphere[positionPriceId - 1] \leftarrow 1$
$cuisine[positionPriceId - 1] \leftarrow 1$
$itemProfile \leftarrow price+payment+alcohol+smookingarea+dresscode+parking+ installation + atmosphere + cuisine$
return $itemProfile$

Algorithm 7 Get recommendations

Require: The currentUser and ratingMatrix.
Ensure: The Top-N list of recommendations for the current user.

Dictionaries $totals \leftarrow \{\}, sumSimilarity \leftarrow \{\}$
$predictions \leftarrow$[ ]
for $otherUser$ to size of $ratingMatrix$ **do**
  if $otherUser = currentUser$ **then**
    jump next $otherUser$
  end if
  $similarityValue \leftarrow$ get pearsonSimilarity
  if $similarityValue \leq 0$ **then**
    jump next $otherUser$
  end if
  for $item$ to size of $profileOther$ **do**
    if $item$ is not in $profileUser$ **then**
      if $profileUser[item] = 0$ **then**
        Set in $totals \leftarrow item$
        $totals[item]$ Add $ratingMatrix[otherUser][item] * similarityValue$
        Set in $sunSimilarity \leftarrow item$
        $sumSimilarity$ Add $similarityValue$
      end if
    end if
  end for
end for
for each $(item, total)$ in $totals$ **do**
  $predictions \leftarrow [(total/sumSimilarity[item], item)]$
end for
Ranking of $predictions$
return $predictions$

Algorithm 8 Get Pearson correlation

Require: The currentUser, otherUser and preferences.
Ensure: The pearsonCorrelation score.

Dictionaries $itemsRatedMutually \leftarrow \{\}$
for each $item$ in preferences of $currentUser$ **do**
  if $item$ is in preferences of $currentUser$ **then**
    jump next $itemsRatedMutually[item] \leftarrow 1$
  end if
end for
$numberElements \leftarrow$ size of $itemsRatedMutually$
if $itemsRatedMutually = 0$ **then**
  return 0
end if
for $item$ to size of $itemsRatedManually$ to get all preferences **do**
  $sumCurrentUser \leftarrow preferences[currentUser][item]$
  $sumOtherUser \leftarrow preferences[otherUser][item]$
end for
for $item$ to size of $itemsRatedManually$ to get squares **do**
  $squareCurrentUser \leftarrow square(preferences[currentUser][item])^2$
  $squareOtherUser \leftarrow square(preferences[otherUser][item])^2$
end for
for $item$ to size of $itemsRatedManually$ to get sum of products **do**
  $sumProduct \leftarrow preferences[currentUser][item] * preferences[otherUser][item]$
end for
$pearsonNumerator \leftarrow sumProduct - ((sumCurrentUser * sumOtherUser)/numberElements)$
$pearsonDenominator \leftarrow square(squareCurrentUser - ((sumCurrentUser)^2/numberElements) * squareOtherUser - ((sumOtherUser)^2/numberElements))$
$pearsonCorrelation \leftarrow pearsonNumerator/pearsonDenominator$
return $pearsonCorrelation$ among two users

Algorithm 9 Matrix factorization

Require: R is a matrix to be factorized, dimension N * M, P an initial matrix of dimension N * K, Q an initial matrix of dimension M * K, K is the number of latent features, steps for the maximum number of steps to perform the optimization, alpha is the learning rate and beta is the regularization parameter.
Ensure: The factorized matrix P and Q.

```
alpha ← 0.0001, beta ← 0.001
QMatrix ← QMatrix ∗ T
for step to rangeSteps do
  for i to size of RMatrix do
    for j to size of RMatrix[i] do
      if RMatrix[i][j] > 0 then
        e_{i,j} ← RMatrix[i][j] − dotProduct(PMatrix[itoend], QMatrix[inittoj])
      end if
      for k to range of KFactors do
        PMatrix[i][k] ← PMatrix[i][k] + alpha ∗ (2 ∗ e_{i,j} ∗ QMatrix[k][j] − beta ∗
        PMatrix[i][k])
        QMatrix[k][j] ← QMatrix[k][j] + alpha ∗ (2 ∗ e_{i,j} ∗ PMatrix[i][k] − beta ∗
        QMatrix[k][j])
      end for
    end for
  end for
  eR ← dotProduct(PMatrix ∗ QMatrix)
  for i to range of RMatrix do
    for j to size of RMatrix[i] do
      if RMatrix[i][j] > 0 then
        e ← e + (beta/2) ∗ PMatrix[i][k]^2 + QMatrix[i][j]^2
      end if
    end for
  end for
  if e < 0 then
    break
  end if
end for
return PMatrix, QMatrix ∗ T
```

Appendix B

USE Questionnaire

Usefulness

- It helps me be more effective.
- It helps me be more productive.
- It is useful.
- It gives me more control over the activities in my life.
- It makes the things I want to accomplish easier to get done.
- It saves me time when I use it.
- It meets my needs.
- It does everything I would expect it to do.

Ease of Use

- It is easy to use.
- It is simple to use.
- It is user friendly.
- It requires the fewest steps possible to accomplish what I want to do with it.
- It is flexible.
- Using it is effortless.
- I can use it without written instructions.
- I don't notice any inconsistencies as I use it.
- Both occasional and regular users would like it.
- I can recover from mistakes quickly and easily.
- I can use it successfully every time.

Ease of Learning

- I learned to use it quickly.
- I easily remember how to use it. It is easy to learn to use it.
- I quickly became skillful with it.

Satisfaction

- I am satisfied with it.
- I would recommend it to a friend.
- It is fun to use.
- It works the way I want it to work.
- It is wonderful.
- I feel I need to have it.
- It is pleasant to use.

Source: From the work of Lund (2001). Note: Users rate agreement with these statements on a 7-point Likert scale, ranging from strongly disagree to strongly agree. Statements in italics were found to weight less heavily than the others.

Bibliography

[1] Réka Albert, Hawoong Jeong, and Albert-László Barabási. Internet: Diameter of the world-wide web. *Nature*, 401(6749):130–131, 1999.

[2] Peter J Angeline. Evolving fractal movies. In *Proceedings of the 1st annual conference on genetic programming*, pages 503–511. MIT Press, 1996.

[3] Sander Bakkes, Chek Tien Tan, and Yusuf Pisan. Personalised gaming: a motivation and overview of literature. In *Proceedings of the 8th Australasian Conference on Interactive Entertainment: Playing the System*, page 4. ACM, 2012.

[4] Shumeet Baluja, Dean Pomerleau, and Todd Jochem. Simulating user"s preferences: Towards automated artificial evolution for computer generated images. 1993.

[5] Shumeet Baluja, Dean Pomerleau, and Todd Jochem. Towards automated artificial evolution for computer-generated images. *Connection Science*, 6(2-3):325–354, 1994.

[6] Peter Bentley. An introduction to evolutionary design by computers. *Evolutionary design by computers*, pages 1–73, 1999.

[7] John Biles. Genjam: A genetic algorithm for generating jazz solos. In *Proceedings of the International Computer Music Conference*, pages 131–131. INTERNATIONAL COMPUTER MUSIC ACCOCIATION, 1994.

[8] John Biles, Peter Anderson, and Laura Loggi. Neural network fitness functions for a musical iga. 1996.

[9] John A Biles. Life with genjam: Interacting with a musical iga. In *Systems, Man, and Cybernetics, 1999. IEEE SMC'99 Conference Proceedings. 1999 IEEE International Conference on*, volume 3, pages 652–656. IEEE, 1999.

[10] John A Biles. Genjam: Evolution of a jazz improviser. *Creative evolutionary systems*, 168:2, 2002.

[11] Terence Broughton, A Tan, and Paul S Coates. The use of genetic programming in exploring 3d design worlds. In *CAAD futures 1997*, pages 885–915. Springer, 1997.

[12] Armelle Brun, Anne Boyer, and Liana Razmerita. Compass to locate the user model i need: Building the bridge between researchers and practitioners in user modeling. In *International Conference on User Modeling, Adaptation, and Personalization*, pages 303–314. Springer, 2010.

[13] Jeff Clune and Hod Lipson. Evolving three-dimensional objects with a generative encoding inspired by developmental biology. In *Proceedings of the European Conference on Artificial Life*, pages 144–148, 2011.

[14] Andy Cockburn and Bruce McKenzie. What do web users do? an empirical analysis of web use. *International Journal of human-computer studies*, 54(6):903–922, 2001.

[15] Sumit Das, Terry Franguiadakis, Michael E Papka, Thomas A DeFanti, and Daniel J Sandin. A genetic programming application in virtual reality. In *International Conference on Evolutionary Computation*, pages 480–484, 1994.

[16] Richard Dawkins. *The blind watchmaker: Why the evidence of evolution reveals a universe without design.* WW Norton & Company, 1986.

[17] N Deepa, S Priyadarsini, VENKATESH Mahadevan, and Joel Wilson. Adaptive hypermedia using link annotation technology and recommender model (ahlarm). *J. Theor. Appl. Inf. Technol*, 46(2):1022, 2012.

[18] Sebastian Deterding. Just add points? what ux can (and cannot) learn from game design. *UXCamp Europe, Berlin, 30 May 2010*, 2010.

[19] Sebastian Deterding, Dan Dixon, Rilla Khaled, and Lennart Nacke. From game design elements to gamefulness: defining gamification. In *Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments*, pages 9–15. ACM, 2011.

[20] Sebastian Deterding, Miguel Sicart, Lennart Nacke, Kenton O'Hara, and Dan Dixon. Gamification. using game-design elements in non-gaming contexts. In *CHI'11 Ex-*

tended Abstracts on Human Factors in Computing Systems, pages 2425–2428. ACM, 2011.

[21] Terry Disz, Michael E Papka, and Rick Stevens. Ubiworld: an environment integrating virtual reality, supercomputing, and design. In *Heterogeneous Computing Workshop, 1997.(HCW'97) Proceedings., Sixth*, pages 46–57. IEEE, 1997.

[22] Qaiser Shehryar Durrani. Cognitive modeling: a domain independent user modeling. In *Systems, Man, and Cybernetics, 1997. Computational Cybernetics and Simulation., 1997 IEEE International Conference on*, volume 1, pages 217–220. IEEE, 1997.

[23] Mark Elsom-Cook. Student modelling in intelligent tutoring systems. *Artificial Intelligence Review*, 7(3-4):227–240, 1993.

[24] Sidney Fels and Jonatas Manzolli. Interactive, evolutionary textured sound composition. In *Multimedia 2001*, pages 153–164. Springer, 2002.

[25] Gerhard Fischer. User modeling in human–computer interaction. *User modeling and user-adapted interaction*, 11(1-2):65–86, 2001.

[26] Dun-wei Gong and Jie Yuan. Impact of individuals' fitness expressions on interactive genetic algorithms' performances. In *2009 Chinese Control and Decision Conference*, pages 2415–2420. IEEE, 2009.

[27] Dun-wei Gong, Jie Yuan, and Xiao-yan Sun. Interactive genetic algorithms with individuals fuzzy fitness. *Computers in Human Behavior*, 27(5):1482–1492, 2011.

[28] Juho Hamari and Veikko Eranti. Framework for designing and evaluating game achievements. In *Proc. DiGRA*, volume 115, pages 122–134. Citeseer, 2011.

[29] Juho Hamari, Jonna Koivisto, and Harri Sarsa. Does gamification work?–a literature review of empirical studies on gamification. In *2014 47th Hawaii International Conference on System Sciences*, pages 3025–3034. IEEE, 2014.

[30] Tom Hargreaves, Michael Nye, and Jacquelin Burgess. Making energy visible: A qualitative field study of how householders interact with feedback from smart energy monitors. *Energy policy*, 38(10):6111–6119, 2010.

[31] Jürgen Heller, Christina Steiner, Cord Hockemeyer, and Albert Dietrich. Competence-based knowledge structures for personalised learning. *International Journal on ELearning*, 5(1):75, 2006.

[32] Trey Hickman. Total engagement: Using games and virtual worlds to change the way people work and businesses., 2010.

[33] Damon Horowitz. Generating rhythms with genetic algorithms. In *AAAI*, volume 94, page 1459, 1994.

[34] Kai Huotari and Juho Hamari. Defining gamification: a service marketing perspective.

In *Proceeding of the 16th International Academic MindTrek Conference*, pages 17–22. ACM, 2012.

[35] W Lewis Johnson, Hannes Högni Vilhjálmsson, and Stacy Marsella. Serious games for language learning: How much game, how much ai? In *AIED*, volume 125, pages 306–313, 2005.

[36] Roula Karam, Piero Fraternali, Alessandro Bozzon, and Luca Galli. Modeling end-users as contributors in human computation applications. In *International Conference on Model and Data Engineering*, pages 3–15. Springer, 2012.

[37] Judy Kay and Gord McCalla. Coming of age: Celebrating a quarter century of user modeling and personalization: Guest editors introduction. *User Modeling and User-Adapted Interaction*, 22(1):1–7, 2012.

[38] Alfred Kobsa. Generic user modeling systems. *User modeling and user-adapted interaction*, 11(1-2):49–63, 2001.

[39] JE Korteling, AS Helsdingen, RR Sluimer, ML van Emmerik, and B Kappé. *Transfer of Gaming: Transfer of training in serious gaming.* Soesterberg: TNO, 2011.

[40] Janaki Kumar. *Gamification at work: Designing engaging business software.* Springer, 2013.

[41] Ik Soo Lim and Daniel Thalmann. Pro-actively interactive evolution for computer animation. In *Computer Animation and Simulation99*, pages 45–52. Springer, 1999.

[42] Ik Soo Lim and Daniel Thalmann. Solve customers' problems: interactive evolution for tinkering with computer animation. In *Proceedings of the 2000 ACM symposium on Applied computing-Volume 1*, pages 404–407. ACM, 2000.

[43] Henrik Hautop Lund, Luigi Pagliarini, and P Miglino. Artistic design with ga and nn. In *Proceedings of the 1st Nordic Workshop on Genetic Algorithms and Their Applications (1NWGA)*, pages 97–105, 1995.

[44] Evelyne Lutton, Emmanuel Cayla, and Jonathan Chapuis. Artie-fract: The artists viewpoint. In *Workshops on Applications of Evolutionary Computation*, pages 510–521. Springer, 2003.

[45] Jon McCormack. Interactive evolution of l-system grammars for computer graphics modelling. *Complex Systems: from biology to computation*, pages 118–130, 1993.

[46] Jane McGonigal. *Reality is broken: Why games make us better and how they can change the world*. Penguin, 2011.

[47] Michael McKenna and David Zeltzer. Dynamic simulation of autonomous legged locomotion. In *ACM SIGGRAPH Computer Graphics*, volume 24, pages 29–38. ACM, 1990.

[48] Michael F McTear. User modelling for adaptive computer systems: a survey of recent developments. *Artificial intelligence review*, 7(3-4):157–184, 1993.

[49] Hyeun-Jeong Min and Sung-Bae Cho. Creative 3d designs using interactive genetic algorithm with structured directed graph. In *Pacific Rim International Conference on Artificial Intelligence*, pages 391–400. Springer, 2004.

[50] Gary Lee Nelson. Sonomorphs: An application of genetic algorithms to the growth and development of musical organisms. In *Proceedings of the Fourth Biennial Art & Technology Symposium*, volume 155, 1993.

[51] Gary Lee Nelson. Further adventures of the sonomorphs. *Conservatory of Music Report, Oberlin College*, 1995.

[52] Thang Nguyen and Thomas Huang. Evolvable 3d modeling for model-based object recognition systems. *Advances in Genetic Programming*, pages 459–475, 1994.

[53] Thang C Nguyen, David E Goldberg, and Thomas S Huang. Evolvable modeling: structural adaptation through hierarchical evolution for 3–d model-based vision. *Urbana and Beckman Institute and Coordinated Science Laboratory, University of Illinois*, 1993.

[54] Chris O'Brien. Get ready for the decade of gamification. *San Jose Mercury News*, 24, 2010.

[55] Gabriela Ochoa. On genetic algorithms and lindenmayer systems. In *International Conference on Parallel Problem Solving from Nature*, pages 335–344. Springer, 1998.

[56] Miho Ohsaki, Hideyuki Takagi, and Kimiko Ohya. An input method using discrete fitness values for interactive ga. *Journal of Intelligent & Fuzzy Systems*, 6(1):131–145, 1998.

[57] T Onisawa, W Takizawa, and M Unehara. Composition of melody reflecting user's feeling. In *Industrial Electronics Society, 2000. IECON 2000. 26th Annual Confjerence of the IEEE*, volume 3, pages 1620–1625. IEEE, 2000.

[58] Larry Page and Sergey Brin. Google. ”http://www.google.com”, 1998.

[59] Michael E Papka and Rick Stevens. Ubiworld: An environment integrating virtual reality, supercomputing and design. In *High Performance Distributed Computing, 1996., Proceedings of 5th IEEE International Symposium on*, pages 306–307. IEEE, 1996.

[60] Ian C Parmee. The concrete arch dam. In *Artificial Neural Nets and Genetic Algorithms*, pages 544–551. Springer, 1993.

[61] Ian C Parmee. Genetic algorithms and hydropower system design. *Computer-Aided Civil and Infrastructure Engineering*, 13(1):31–41, 1998.

[62] MMM Peeters, Karel van den Bosch, JJ Meyer, and Mark A Neerincx. Situated cognitive engineering: the requirements and design of automatically directed scenario-based training. In *ACHI 2012, The Fifth International Conference on Advances in Computer-Human Interactions*, pages 266–272. XPS: Xpert Publishing Services, 2012.

[63] Riccardo Poli and Stefano Cagnoni. Genetic programming with user-driven selection: Experiments on the evolution of algorithms for image enhancement. *Genetic Programming*, pages 269–277, 1997.

[64] Frédéric Raynal, Evelyn Lutton, Pierre Collet, and Marc Schoenauer. Manipulation of non-linear ifs attractors using genetic programming. In *Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*, volume 2. IEEE, 1999.

[65] Liana Razmerita. User modeling and personalization of advanced information systems. 2009.

[66] Liana Razmerita, Thierry Nabeth, and Kathrin Kirchner. User modeling and attention support: Towards a framework of personalization techniques. *of: CENTRIC*, page 37, 2012.

[67] Duncan Rowland and Frank Biocca. Evolutionary co-operative design between human and computer: implementation of the genetic sculpture park. In *Proceedings of the fifth symposium on Virtual reality modeling language (Web3D-VRML)*, pages 75–79. ACM, 2000.

[68] Jimmy Secretan, Nicholas Beato, David B D Ambrosio, Adelein Rodriguez, Adam Campbell, and Kenneth O Stanley. Picbreeder: evolving pictures collaboratively online. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1759–1768. ACM, 2008.

[69] Karl Sims. *Artificial evolution for computer graphics*, volume 25. ACM, 1991.

[70] Karl Sims. Interactive evolution of dynamical systems. In *Toward a practice of autonomous systems: Proceedings of the first European conference on artificial life*, pages 171–178, 1992.

[71] Karl Sims. Evolving virtual creatures. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 15–22. ACM, 1994.

[72] Rod Sims. Interactivity: A forgotten art? *Computers in Human Behavior*, 13(2): 157–180, 1997.

[73] John D Sutter. Browse the web, earn points and prizes, 2010.

[74] Hideyuki Takagi. Interactive evolutionary computation: System optimization based on human subjective evaluation. In *IEEE Int. Conf. on Intelligent Engineering Systems (INES98)*, pages 17–19, 1998.

[75] Hideyuki Takagi and Kimiko Ohya. Discrete fitness values for improving the human interface in an interactive ga. In *Evolutionary Computation, 1996., Proceedings of IEEE International Conference on*, pages 109–112. IEEE, 1996.

[76] CM Tam and W Leung. Genetic algorithm modeling aided with 3d visualization in optimizing construction site facility layout. *International council for research and innovation in building and construction*, 5(4):125–36, 2002.

[77] Eran Toch, Yang Wang, and Lorrie Faith Cranor. Personalization and privacy: a survey of privacy risks and remedies in personalization-based systems. *User Modeling and User-Adapted Interaction*, 22(1-2):203–220, 2012.

[78] Stephen Todd. Artificial life or surreal art? In *1st European Conf. on Artificial Life*, pages 504–513. MIT Press, 1992.

[79] Stephen Todd and William Latham. Evolutionary art and computers. 1994.

[80] Stephen Todd and William Latham. The mutation and growth of art by computers. *Evolutionary design by computers*, 9:221–250, 1999.

[81] Nao Tokui and Hitoshi Iba. Music composition with interactive evolutionary computation. 2000.

[82] Tatsuo Unemi. Sbart 2.4: an iec tool for creating 2d images, movies, and collage. In *Proceedings of the Genetic and Evolutionary Computation Conference Program, Las Vegas*, 2000.

[83] Tatsuo Unemi. Sbeat3: A tool for multi-part music composition by simulated breeding. In *Proceedings of the eighth international conference on artificial life. MIT Press, Cambridge*, pages 410–413, 2003.

[84] Jeffrey Ventrella. Explorations in the emergence of morphology and locomotion behavior in animated characters. In *Artificial Life IV*, pages 436–441, 1994.

[85] Jeffrey Ventrella. Disney meets darwin-the evolution of funny animated figures. In *Computer Animation'95., Proceedings.*, pages 35–43. IEEE, 1995.