Modeling Human Interactions in Collaborative Interactive Evolutionary Computation

Anonymized Version

Ben Trovato
Institute for Clarity in Documentation
P.O. Box 1212
Dublin, Ohio 43017-6221
trovato@corporation.com

G.K.M. Tobin
Institute for Clarity in Documentation
P.O. Box 1212
Dublin, Ohio 43017-6221
webmaster@marysville-ohio.com

Lars Thørväld
The Thørväld Group
1 Thørväld Circle
Hekla, Iceland
larst@affiliation.org

Lawrence P. Leipuner
Brookhaven Laboratories
P.O. Box 5000
lleipuner@researchlabs.org

ABSTRACT

The necessary intervention of humans in interactive evolutionary computational systems has inherent drawbacks arising from the very nature of the algorithms, namely, the human fatigue caused by the interaction, and the boredom arising when users evaluate a large number of artifacts. To tackle these issues, in this paper we propose a human-centered framework to model complex interactions on these systems. A case study is presented where the model is applied in the development of a collaborative evolutionary interactive system. Both conceptual and implementation details are provided where the technique is used to measure and increase user engagement and participation. Our experiments show that the model can be successfully applied in a gamification technique developed to increase user engagement, which implies that this technique can successfully be used to decrease user fatigue and boredom, and thus increase the *performance* of the interactive system.

CCS CONCEPTS

•Human-centered computing → Collaborative and social computing systems and tools; •Computing methodologies → *Heuristic function construction;*

KEYWORDS

Interactive evolutionary computation, Human Centered Computing

ACM Reference format:
Ben Trovato, G.K.M. Tobin, Lars Thørväld, and Lawrence P. Leipuner. 2017. Modeling Human Interactions in Collaborative Interactive Evolutionary Computation. In *Proceedings of the Genetic and Evolutionary Computation Conference 2017, Berlin, Germany, July 15–19, 2017 (GECCO '17),* 8 pages.
DOI: 10.475/123_4

DOI: 10.475/123_4

1 INTRODUCTION

Interactive evolutionary computation (IEC) systems are, in general, evolutionary methods whose fitness evaluations are performed by humans through an interactive system [12]. They are usually applied in problems where the fitness function is not known or simply does not exist, and the result of optimization should fit a certain human need or desire such as an aesthetic ideal. That is why their use cases include the evolution of objects with subjective characteristics, such as visual appeal and attractiveness [7] as well as others where human behavior is considered, for instance the optimization of teamwork [24] or creativity [49]. In the cases when human interaction is responsible of other aspects of the evolutionary process, these IEC methods are classified by some authors as human-based evolutionary computation [23] or as human-based computation [36].

IEC systems are an interesting venue of research, since they have demonstrated their ability for effectively producing art and design [3, 25, 41, 46], as well as other types of artifacts in many other domains [45]. However, the necessary intervention of humans brings certain challenges to designers of IEC methods; namely, human evaluations are scarce, slow and expensive, there is human fatigue caused by the interaction [45], and also boredom arises when users evaluate a large number of phenotypes, many of which are not interesting or are very similar to each other. Moreover, the performance of these systems effectively depends on the number of users they are able to include; to reach more users, IEC systems are some times developed as web applications depending on visitors to help with the search, using both anonymous and registered users. Some systems employ a collaborative technique, where several users participate in the evaluation, this method is called Collaborative-IEC (C-IEC) [39, 40, 47]. Including an C-IEC in a volunteer system can lower the requirements for participants in the experiment thus increasing the *performance* of the whole system. But using a volunteer based system raises other issues [37, 48], such as the volunteer's lack of accountability, and the need to build trust between participants and project owners. Other issues of interest for project owners are also the difficulty of predicting the amount of time and resources a volunteer is willing to spend on the system, and how they decide if they participate or not [32].

In order to increase volunteer participation and to tackle some of the issues mentioned above, we proposed a software framework following a human centered design [19], giving extensive attention to volunteers, not only because their explicit evaluation is essential, but also because the context of the interaction affects the system as a whole.

For example, in a C-IEC application fitness assignment depends on the actions of a social network of users. These actions are triggered when they tag, share, rate, store or delete a phenotype. Then, the selection of parents could depend on the previous actions, leveraging information, such as the fact that they both have the same tag, or were shared by similar users.

Data available from the interaction is also used to increase the engagement of users in the system by applying gamification techniques. Gamification is a technique defined by Deterding et al. [10] as

> the use of game design elements in non-game contexts.

The gamification element employed in this work is a rewarding mechanism [11]. In general rewards consist of a reputation system with score points, levels and leader boards. Points are awarded to users in response of the accomplishment of certain activities that need to be encouraged. Levels depend on the score and certain features of the game are only available to gamers when they reach a giving level.

The development of a web based C-IEC application for evolving artistic drawings is described to give the reader details about the utilization of the proposed framework. Then, a rewarding mechanism is implemented in the same application to study the changes in participation when applying a gamification technique. Finally, an experiment is conducted comparing three versions of the application: one with out gamification and two employing a different gamification technique.

The rest of this paper is organized as follows. Section 2 presents related work on the topic of collaborative interactive evolution. Then, Section 3 presents the human centered framework for C-IEC applications which the main proposal of this work. Implementation details of the data model are presented in Section 4. Next the EvoDrawings application case study is presented in Section 5. The experimental set-up is described in Section 6, an the results are discussed in Section 7. Finally, concluding remarks are provided in Section 8.

2 RELATED WORK

An early example of C-IEC is the Galapagos Project [42], an exhibit in the Tokyo Multimedia Museum (1997–2000) were visitors interacted with images presented in twelve displays by selecting those they found most aesthetically interesting by standing on step sensors in front of them. Web based systems were introduced later, with Langdon's system [26] which evolved fractal representations of virtual creatures. Similarly, Secretan et al. [39] and Clune and Lipson [6] use web-based IEAs to evolve artistic artifacts using a generative encoding.

Some C-IEC systems promote user engagement by presenting interesting information to users, for instance the genetic lineage of each phenotype or the most popular or best rated solution [6, 39]. An example is the recent work by Wagy & Bongard [47] where user interaction is needed for evaluating fitness and developing new designs of robot locomotion. Collaboration is encouraged by gamifying the system using the maximum distance indicator to inspire the user to try and "beat" previous designs. In any case, using gamification techniques imply dealing with IEC systems as socio-technical constructs, where the social aspects are essential to understand its dynamics. In this sense, conclusions reached with other systems such as NodEO [31] can also be applied to these systems; and applying social network techniques such as graph analysis to their study will allow us to understand them more thoroughly.

Given the human fatigue limitation when applying IEAs, some authors have tried to mitigate the problem by allowing the algorithm to collaborate with the user, so that sometimes users perform the evaluation, but also specific measures are included into the algorithm to perform the evaluation of some features automatically. For instance, Reis et al. added some terrain measures (such as accessibility and edge length) to a standard IEC in [13]. This way the algorithm was capable of providing terrains that would otherwise have needed the users' evaluation for these specific features. Seyama and Munetomo [40] also propose the reduction of user fatigue by using a collaborative filtering algorithm to show only the information utilized by similar users as they collaborate with a large number of users for the interactive modeling of 3D glasses.

Promoting user engagement and reducing user fatigue requires the management of the complex interactions found in these systems. Our approach is proposed next.

3 HUMAN-CENTERED C-IEC FRAMEWORK

The general goal of this research is to develop a human-centered [18] software framework that can be used to increase volunteer participation in C-IEC systems. A framework is defined as a reusable architectural design together with an implementation [4], in this case providing generalized components to developers and researchers of C-IEC systems. The proposed framework includes components that can be refined to increase participation and also to minimize the amount of evaluations needed for the evolutionary process in a given IEC application. Software frameworks often have a vision [5] guiding their design. Before diving into details the main design considerations of the framework are explained next:

- **Users are human.** The framework follows the approach of human centered computing [38], in which the context, environment, interfaces, preferences, accessibility, human relations, cognitive limitations, culture, creativity and other human aspects are an integral part of the system. Humans are the computing resources of the system, having unique characteristics as those identified by Sun & Dance [44]:
  (1) humans can solve computer hard problems;
  (2) humans are very good at exception handling,

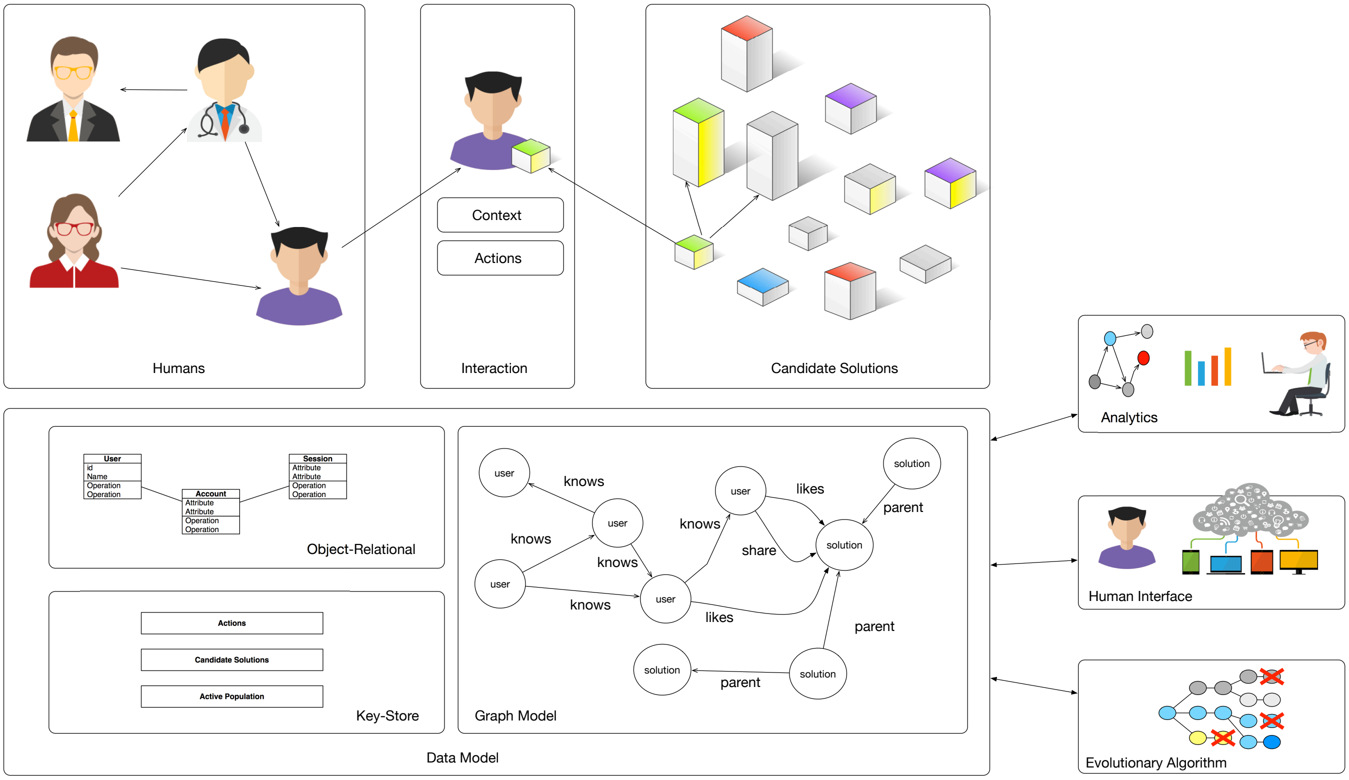

Figure 1: IEC Human-centered framework.

(3) humans have creativity, (4) humans have cognitive load limitation, (5) humans are vulnerable to psychological manipulation, (6) humans are prone to errors, especially for reflective tasks.

- **Users are volunteers.** Users donate their computing resources, so they are unaccountable and sometimes they try to game the system. Project owners must actively promote and design the interactive system to engage volunteers [35].
- **Users are not alone** Relationships between users in an interactive evolutionary algorithm can be modeled as a social network, with well established semantics, algorithms and metrics [1]. A graph model could enable researchers to find other ways of identifying leaders of opinion or measuring the similarity between user's preferences. These measures can then be used by recommender algorithms selecting phenotypes according to their preferences.
- **Context of use matters.** Fischer [14] defines context as

  the interaction between humans and computers in socio-technical systems that takes place in a certain context referring to the physical and social situation in which computational devices and environments are embedded.

  Fischer also identifies the important aspects to consider when the context is used: how is contextual data obtained, how is context represented and what goals and purposes the context has when it is used in a particular application. An IEC system will be used within a certain range of technical, physical and social or organizational environments [28] that may affect its use.
- **Interaction is a stream of actions.** Real time processing of users' actions could be needed for certain applications when data is captured by sensors, or directly captured as user input. For example, social networks encourage users to publish their interactions with other users, media objects and places. Users of social networks (for instance the Facebook Graph) are accustomed to express these complex relationships in sentences such as: "John and Ann eating breakfast at Tony's". As this is gaining accptence, there are iniciatives like the W3C Activity Streams 2.0 specification, used for representing common activities in social web applications [43].

The above considerations have guided the design of the framework, and throughout they have been treated as application constraints. In order to satisfy theses requirements the human-centered C-IEC architecture consists of three high level components depicted in figure 1:

(1) **Interactive System.** This is the real world system that we are going to represent in the data model, it consist of human users and their interactions with one or more phenotypes from the population. There are many ways in which humans could interact with these phenotypes. The interaction consists of a set of actions and takes place in

a certain context; for example, through a mobile device or by interacting with real world objects [8, 9]. There is also the possibility that fitness or even part of the search is done by devices lent by humans [31]. The information gathered through the interaction is the primary focus as it will guide the search.

(2) **Data Model.** A data model is used to describe the IEC system prior to a physical implementation. Depending on the domain several models can be used, these options and their physical implemementations are explained in Section 4.

(3) **Human Interface.** The appropriate human interface will depend on the application domain, the current framework implementation employs a web based application. Developers can use different templates, to present one or more phenotypes at a time, and the type of rating system: based on "likes" or in a rating from one to five (stars). Other implementations do not need a graphical interface at all, using sensors and actuators.

(4) **Evolutionary Algorithm.** The Evolutionary Algorithm (EA) algorithm interacts with the data model. The EA is decoupled from the data model, the algorithm could be implemented internally, use an external library, or even be human based evolution, where volunteers select the parents of the next generation and upload the new individuals as in the XYZ project [8].

(5) **Analytics.** Researchers and deveopers will need to query the data model in order to analyzing the system's behaviour.

4 DATA MODEL IMPLEMENTATION

In this section, the core component of the framework is presented. Three database models are employed to store different elements of the system, next these are described including the technology selected for their physical implementation:

- **EvoSpace-Redis.** EvoSpace is a population store [16] for the development of evolutionary algorithms that are intended to run on a cloud computing model. The population is decoupled from any particular evolutionary algorithm. Candidate solutions are stored as of objects in a population, and they can be withdrawn, processed and replaced using a specified set of methods [17]. The population is stored in-memory, using the Redis key-value database. Redis was chosen over a relational database management system because it provides a hash based implementation of sets and queues which are natural data structures for the EvoSpace model. Basically, a sample of candidate solutions are retrieved from the server, evaluated and then sent back. The same operations are used to evolve the population. In EvoSpace individuals replaced in the population are stored indefinitely, to permit users to store permalinks to them. Implementation details are presented in [15].
- **Graph-Neo4J.** Collaborative IEC (C-IEC) systems need to store highly connected data, as it is common in current applications like social networks. In order to deal with large datasets of connected data found in these systems, graph databases [2] have been proposed as an alternative to relational databases which have performance limitations when dealing with highly connected data [21]. A graph is proposed for modeling the social network of users, their interaction with candidate solutions, and the relationships between them in the population. The graph database system used in the implementation is Neo4J, which is a scalable solution [21, 34], well supported and documented in PaaS infrastructures like Heroku. The Cypher query language it used to retrieve views from the graph. An example query is shown in Figure 2 where the relations between users and solutions are presented.
- **Relational Objects-PostgreSQL.** The PostgreSQL relational database system is also employed because user sessions and authentication, as well as dynamic web pages are handled directly by the Django web framework [15].

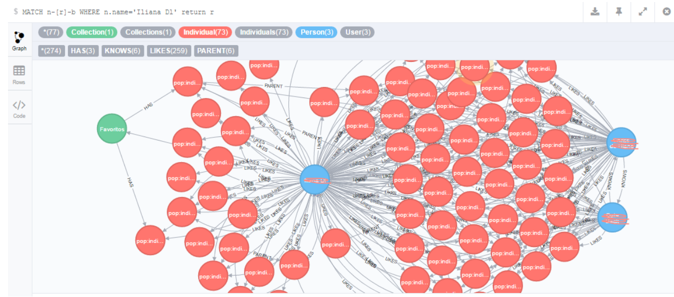

Figure 2: Example query in Cypher.

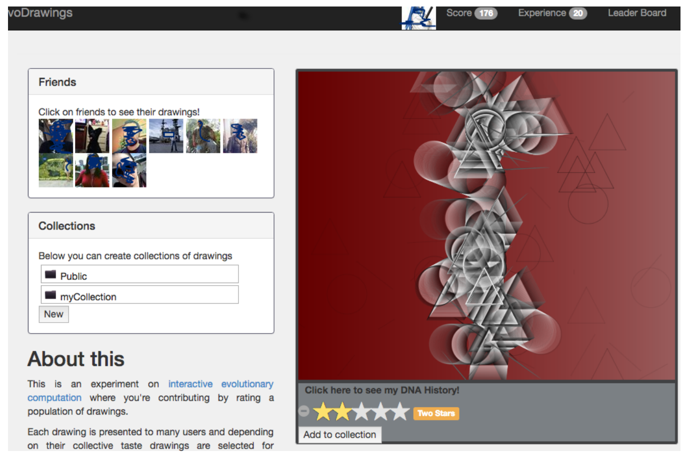

Figure 3: User interface of the EvoDrawings application. (ANONIMIZED IN THIS VERSION)

5 CASE STUDY: EVODRAWINGS

As a case study, a C-IEC application was developed by extending the EvoSpace-Interactive (ES-I) platform [15] A brief description of the application is presented next.

5.1 Collaboration

Users need to authenticate themselves to the system using their Facebook account. In that sense, some users might not be interested either because they do not want to give that information or simply because they do not use that social network. Even if we might lose some users that way, the additional information we obtain for scoring phenotypes more than balances that. After logging in, users can collaborate with their Facebook friends, sharing the phenotypes they like, or by taking phenotypes from their friend's collections by using the web interface depicted in Figure 3. At the top left corner a list of Facebook friends is presented to encourage users to interact with the system. In the central *Wall* area, a phenotype sampled from the population that is being evolved via the evolutionary algorithm is shown to the user. Here, the user can interact with the system in two ways. First, he can assign a rating to the phenotype or choose to add an image to one of their *Collections*. A collection is a special folder that stores those phenotypes a user likes and wishes to save. After the user finishes interacting with the phenotype on the Wall, he can choose to retrieve a new one from the population. At the left hand side of Figure 3, the web page shows the *Collections* section. The user can create several collections, to group and organize his favorite artifacts. Moreover, users can browse the content of each collection and from there share images through their social network. This makes the assignment of fitness through the rating system a *social* activity, pursuing the objective of this work, which is to increase user engagement.

5.2 Graph Model

The graph model for the EvoDrawings application has the following types of nodes: **User**, **Phenotype** and **Collection**. The Collection node represents a collection of drawings belonging to a user. One collection can contain many phenotypes or be empty. A single phenotype could be shared by many collections. The interaction between these entities are represented by the following edges:

- **Likes** This relation describes the interaction between a user and a phenotype in which a rating value is assigned.
- **Knows** The relation connects two users that know each other in the Facebook social graph.
- **Parent** Describes what phenotype is the parent of a new phenotype.
- **Has** The relation describes an ownership relation between users and those collections they own.

5.3 Gamification

The rewarding mechanism as it is applied in EvoDrawings gives more importance to the preference of those users with higher reputation as given by their score points and experience levels. Each time a user does on of these actions their score is incremented by one: start a session, rate a phenotype, create a collection, save a phenotype of the wall to a collection, save a phenotypes from a friend's collection, and explore collections of other friends.

Two variables are used to determine the weight of a user's preference:

- **Experience**: This variable depends on the score and is a value between 0 and 100. A new user starts at zero, and the experience increases until it reaches 100 actions. It is assumed for this case, that once a user reaches this value, it has enough experience on using the application.
- **Participation**: This variable is simply the degree of the user node in the graph (number of edges).

Table 1: Parameters for experiments.

Parameter	Value
Initial Population Size	80
Sample Size	1
Step Size	8 Samples
Mutation	
Selection	Tournament
Tournament Size	6

6 EXPERIMENTAL SETUP

Three versions of EvoDrawings were compared:

- Base (B): All users have the same weight.
- Non Graph Gamification (G): Only experience is considered.
- Graph Gamification (GG): Both experience and participation are considered.

When gamification was employed, all score values known where presented to users and a ranking of users by weight was shown in a window. Table 1 shows the parameters used for the evolutionary algorithm.

At the start of every experiment, a call to participation was issued through social networks. The link used in the call for participation was a shortened Google URL, that provides meta-data and analytics for the users that click on it. In Table 2 the URL for each deployment in the Heroku platform is shown, along with the short URL and the analytics link. In the same table a link to the GitHub application repository used to deploy to Heroku is also listed (ANONIMIZED IN THIS VERSION). Only data for the first week of deployment was considered for the experiments, and they where conducted between January and May of 2016.

7 RESULTS

Before release, each deployment was first tried with a few beta testers. When applying the leader board gamification technique for the first time a problem was found: some users were cheating by giving a rating to an animation even before it was returned from the server, this was done by just constantly clicking the mouse button. This is a common problem found in systems using leader boards because by making the scores visible to other players they are encouraged to compete [20]. The version used in experiments disabled the button until the drawing animation was over. The results of each of the three experiments in terms of participation are detailed next.

Table 2 shows the total number of volunteers, nodes and edges in the graph after each experiment. Moreover, the total number of evaluated phenotypes for each volunteer is presented in figure 4 where users are ranked by the number of phenotypes they rated. In the x axis is the rank and in the y axis the number of phenotypes

Table 2: After a week of the announcement the total number of volunteers, nodes and edges in the graph and analytics URLs

Deployment	Users	Nodes	Edges	URL
B	53	595	2220	goo.gl/jLis4Q.info
G	54	648	2596	goo.gl/jqjNy5.info
GG	68	932	3594	goo.gl/J8TCe1.info

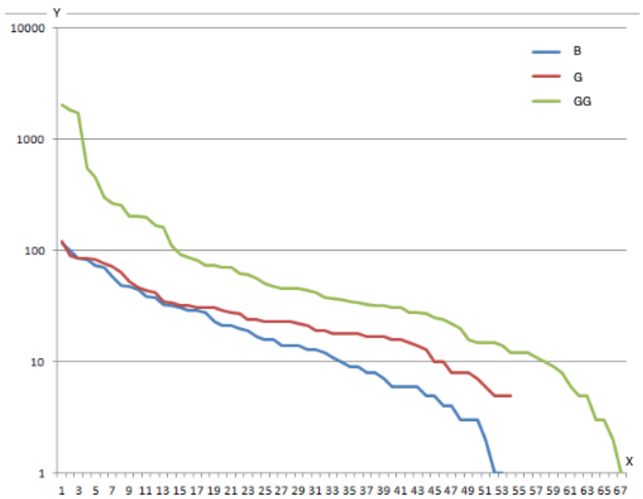

Figure 4: Users ranked by the number of phenotypes they rated vs. the number of phenotypes evaluated using a logarithmic scale.

evaluated using a logarithmic scale. Results show that when considering deployments B and G, the difference came with users with a medium level of participation. When comparing all the experiments the deployment GG had the higher number of participation, besides attracting also the higher number of users.

Moreover, modeling the performance of an C-IEC system involves understanding its dynamics. Previous works on browser-based volunteer computing have used basic metrics such as the number of users or the time spent by every one in the computation [27, 33]. While works on other platforms such as SETI@home [22] have found that the Weibull, log-normal, and Gamma distributions where viable models of the the availability of computing resources in several clusters, which is in concordance with the results obtained in [30], and also browser-based volunteer evolutionary systems like NodIO [31]. The shape of those distributions is a skewed bell with more resources in the *low* areas than in the high areas; i.e., there are many users that give a small amount of cycles, while there are just a few that give many cycles. In order to assess gamification techniques alongside C-IEC follows the same pattern, participation was fitted to a Weibull distribution, and shown in Figures 5 to 7, confirming this a model for user interaction, although with different fitted values in each case. A graph query was used to compare the number of volunteers in deployments B and GG (shown in figures 8 and 9 respectively) showing additional patterns on both social networks.

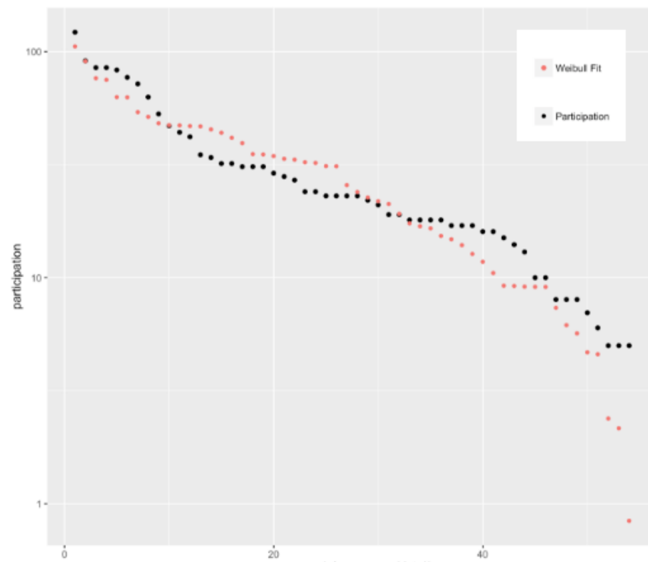

Figure 5: Deployment B. Weibull Fit.

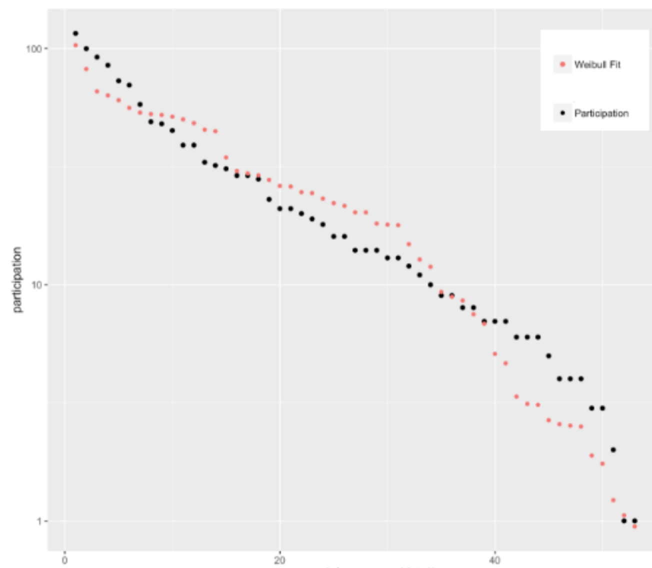

Figure 6: Deployment G. Weibull Fit.

8 CONCLUSIONS

Volunteer-based and C-IEC systems involve the dynamic interaction of many entities and artifacts. Employing a human-centered approach will allow researchers to understand and visualize this kind of systems better.

In this paper a human-centered software framework was proposed, and was validated through the implementation and refinement of a C-IEC application. This framework enabled the implementation of a gamification technique to improve engagement in a case study which provides a common arena where users are aware of the activities of other users in the social neighborhood.

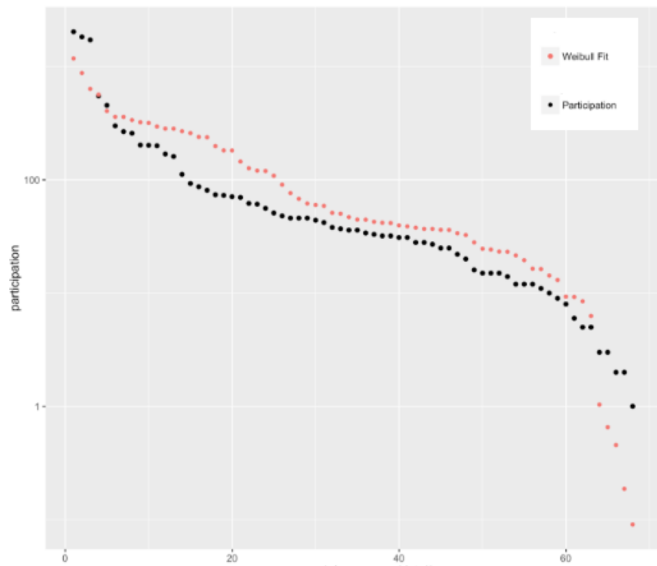

Figure 7: Deployment GG. Weibull Fit.

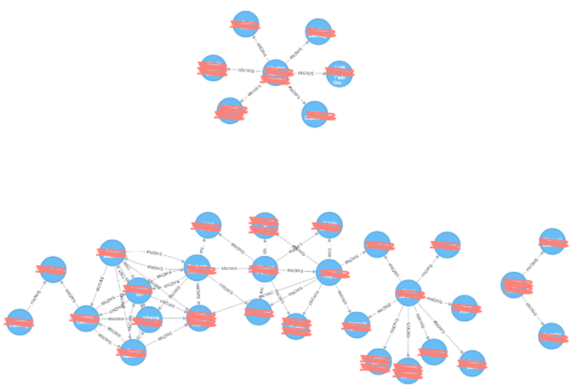

Figure 8: User network view of the Base deployment (B). (ANONIMIZED IN THIS VERSION)

In concordance with the results obtained in other browser-based volunteer systems, after applying the gamification techniques, user participation follows the same pattern and it was fitted to a Weibull distribution.

One of the interesting future lines of work would be to look a bit more closely at the behavior of users as they are rating artifacts in the web system. These initial experiments hint at a possible power law, which might indicate that the IEC system could be self-organizing, a process that would allow it to reach a critical state, as has been found in software repositories, for instance [29]. The dynamics of this kind of system are fundamentally different, and our future research will include exploring these aspects of the system.

Another line of work would be to study the possible negative effects of using gamification techniques to improve engagement, like cheating or literally *gaming* the system to defeat competition. We already found some hints of this behavior at the beginning of the release of this system, but more subtle effect could be taking place. Finally, the refinement of the proposed Human-Centered framework will need more case studies and further multi-disciplinary research.

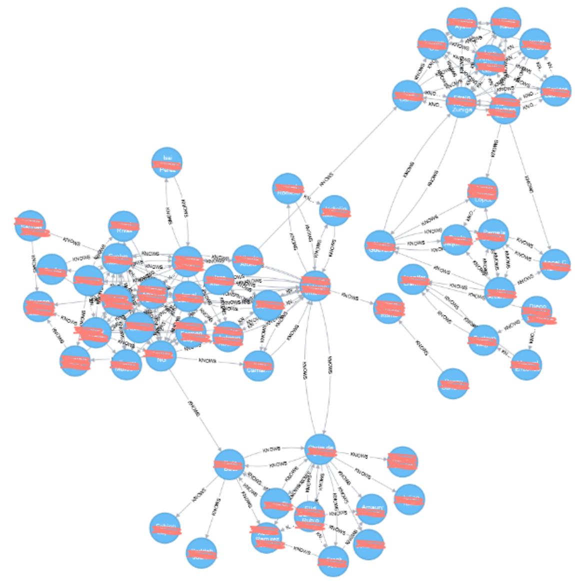

Figure 9: User network view of the Graph Gamification deployment(GG) (ANONIMIZED IN THIS VERSION) .

ACKNOWLEDGMENTS

Reserved
Space for acks.

REFERENCES

[1] Ravindra K Ahuja, Thomas L Magnanti, and James B Orlin. 1993. *Network flows: theory, algorithms, and applications.* Prentice hall, USA.
[2] Renzo Angles. 2012. A comparison of current graph database models. In *Data Engineering Workshops (ICDEW), 2012 IEEE 28th International Conference on.* IEEE, USA, 171–177.
[3] Peter Bentley. 1999. An introduction to evolutionary design by computers. In *Evolutionary Design by Computers*, Peter J. Bentley (Ed.). Morgan Kaufman, San Francisco, USA, Chapter 1, 1–73.
[4] Roy H. Campbell, Nayeem Islam, Ralph Johnson, Panos Kougiouris, and Peter Madany. 1991. Choices, frameworks and refinement. In *Object Orientation in Operating Systems, 1991. Proceedings., 1991 International Workshop on.* IEEE, USA, 9–15.
[5] Cloves Carneiro and Rida Al Barazi. 2010. *Introducing the Rails Framework.* Springer, Germany, Chapter 1, 1–12.
[6] Jeff Clune and Hod Lipson. 2011. Evolving three-dimensional objects with a generative encoding inspired by developmental biology. In *Proceedings of the European Conference on Artificial Life.* MIT press, Boston, MA, USA, 144–148.
[7] Richard Dawkins. 1996. *Climbing Mount Improbable.* W.W. Norton & Company, New York.
[8] F Fernández de Vega, Tania Gallego, Mario García-Valdez, Luis Espada, Lilian Navarro, and V Albarrán. 2014. When artists met EvoSpace-i. In *Evolutionary Computation (CEC), 2014 IEEE Congress on.* IEEE, USA, 2282–2289.
[9] F Fernandez de Vega, Lilian Navarro, Cayetano Cruz, F Chavez, Luis Espada, Patricia Hernandez, and Tania Gallego. 2013. Unplugging evolutionary algorithms: on the sources of novelty and creativity. In *Evolutionary Computation (CEC), 2013 IEEE Congress on.* IEEE, USA, 2856–2863.

[10] Sebastian Deterding, Dan Dixon, Rilla Khaled, and Lennart E Nacke. 2011. From Game Design Elements to Gamefulness: Defining Gamification. In *Proceedings of MindTrek 2011*. ACM, Tampere, Finland, 9–15. DOI: http://dx.doi.org/10.1145/2181037.2181040
[11] Daniel J Dubois and Giordano Tamburrelli. 2013. Understanding gamification mechanisms for software development. In *Proceedings of the 2013 9th Joint Meeting on Foundations of Software Engineering*. ACM, NY, 659–662.
[12] AE Eiben and JE Smith. 2015. Interactive evolutionary algorithms. In *Introduction to Evolutionary Computing*. Springer, Germany, 215–222.
[13] Miguel Frade, Francisco Fernández de Vega, and Carlos Cotta. 2012. Automatic evolution of programs for procedural generation of terrains for video games - Accessibility and edge length constraints. *Soft Comput.* 16, 11 (2012), 1893–1914. DOI: http://dx.doi.org/10.1007/s00500-012-0863-z
[14] Fischer G. 2012. Context-Aware systems-the right information, at the right time, in the right place, in the right way, to the right person. In *AVI fi12 proceedings, Capri Island, Italy*. ACM, USA, 287–294.
[15] Mario García-Valdez, Leonardo Trujillo, Francisco Fernández de Vega, Juan Julián Merelo Guervós, and Gustavo Olague. 2013. Evospace-interactive: A framework to develop distributed collaborative-interactive evolutionary algorithms for artistic design. In *Evolutionary and Biologically Inspired Music, Sound, Art and Design*. Springer, Germany, 121–132.
[16] Mario García-Valdez, Leonardo Trujillo, Francisco Fernández de Vega, Juan J. Merelo Guervós, and Gustavo Olague. 2013. EvoSpace: A Distributed Evolutionary Platform Based on the Tuple Space Model. In *Applications of Evolutionary Computation*, AnnaI. Esparcia-Alcázar (Ed.). Lecture Notes in Computer Science, Vol. 7835. Springer Berlin Heidelberg, Germany, 499–508. DOI: http://dx.doi.org/10.1007/978-3-642-37192-9_50
[17] Mario García-Valdez, Leonardo Trujillo, Juan-J Merelo, Francisco Fernández de Vega, and Gustavo Olague. 2015. The EvoSpace Model for Pool-Based Evolutionary Algorithms. *Journal of Grid Computing* 13, 3 (2015), 329–349. DOI: http://dx.doi.org/10.1007/s10723-014-9319-2
[18] Susan Gasson. 2003. Human-centered vs. user-centered approaches to information system design. *JITTA: Journal of Information Technology Theory and Application* 5, 2 (2003), 29.
[19] Esther Semsei Greenhouse. 2012. Human-centered design. (2012). Retrieved from Livable New York: http://www.aging.ny.gov/LivableNY/ResourceManual/DemographicAndSocialTrendsI.
[20] Trey Hickman. 2010. Total engagement: Using games and virtual worlds to change the way people work and businesses. (2010).
[21] Florian Holzschuher and René Peinl. 2013. Performance of graph query languages: comparison of cypher, gremlin and native access in Neo4j. In *Proceedings of the Joint EDBT/ICDT 2013 Workshops*. ACM, USA, 195–204.
[22] Bahman Javadi, Derrick Kondo, J-M Vincent, and David P Anderson. 2009. Mining for statistical models of availability in large-scale distributed systems: An empirical study of SETI@home. In *Modeling, Analysis & Simulation of Computer and Telecommunication Systems, 2009. MASCOTS'09. IEEE International Symposium on*. IEEE, USA, 1–10.
[23] A. Kosorukoff. 2001. Human based genetic algorithm. In *Systems, Man, and Cybernetics, 2001 IEEE International Conference on*, Vol. 5. IEEE, USA, 3464–3469 vol.5. DOI: http://dx.doi.org/10.1109/ICSMC.2001.972056
[24] Alexander Kosorukoff and David Goldberg. 2002. Evolutionary computation as a form of organization. In *Proceedings of the 4th Annual Conference on Genetic and Evolutionary Computation*. Morgan Kaufmann Publishers Inc., US, 965–972.
[25] T. Kowaliw, A. Dorin, and J. McCormack. 2012. Promoting Creative Design in Interactive Evolutionary Computation. *Evolutionary Computation, IEEE Transactions on* 16, 4 (2012), 523 –536.
[26] W. B. Langdon. 2004. Global Distributed Evolution of L-Systems Fractals. In *Genetic Programming, Proceedings of EuroGP'2004 (LNCS)*, Maarten Keijzer, Una-May O'Reilly, Simon M. Lucas, Ernesto Costa, and Terence Soule (Eds.), Vol. 3003. Springer-Verlag, Berlin, 349–358.
[27] Juan Luís Jiménez Laredo, Pascal Bouvry, D. L. González, Francisco Fernández de Vega, Maribel García-Arenas, Juan J. Merelo-Guervós, and Carlos M. Fernandes. 2014. Designing robust volunteer-based evolutionary algorithms. *Genetic Programming and Evolvable Machines* 15, 3 (2014), 221–244. DOI: http://dx.doi.org/10.1007/s10710-014-9213-5
[28] Maguire M. 2001. Context of use within usability activities. *International Journal of Human-Computer Studies* 55, 4 (2001), 453–483.
[29] Juan J. Merelo. 2016. *Quantifying activity through repository mining: the case of Moose.* Technical Report. GeNeura/UGR/CITIC. 10.6084/m9.figshare.3756768.v3
[30] J. J. Merelo, P.A. Castillo, J.L.J. Laredo, A. Mora, and A. Prieto. 2008. Asynchronous Distributed Genetic Algorithms with JavaScript and JSON. In *WCCI 2008 Proceedings*. IEEE Press, USA, 1372–1379. http://atc.ugr.es/I+D+i/congresos/2008/CEC_2008_1372.pdf
[31] Juan J. Merelo, Pedro A. Castillo, Pablo García-Sánchez, Paloma de las Cuevas, Nuria Rico, and Mario García Valdez. 2016. Performance for the Masses: Experiments with A Web Based Architecture to Harness Volunteer Resources for Low Cost Distributed Evolutionary Computation. In *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference, Denver, CO, USA, July 20 - 24, 2016*, Tobias Friedrich, Frank Neumann, and Andrew M. Sutton (Eds.). ACM, USA, 837–844. DOI: http://dx.doi.org/10.1145/2908812.2908849
[32] J. J. Merelo, Paloma de Las Cuevas, Pablo García-Sánchez, and Mario García-Valdez. 2016. The human in the loop: volunteer-based metacomputers as a socio-technical system. In *Proceedings of the Artificial Life Conference 2016 (Complex Adaptive Systems)*. The MIT Press, One Rogers Street Cambridge MA 02142-1209, Article 103, 8 pages. DOI: http://dx.doi.org/10.7551/978-0-262-33936-0-ch103
[33] J.-J. Merelo, M. García-Valdez, P. A. Castillo, P. García-Sánchez, P. de las Cuevas, and N. Rico. 2016. *NodIO, a JavaScript framework for volunteer-based evolutionary algorithms : first results.* Technical Report. GeNeura/UGR/CITIC. arXiv:cs.DC/1601.01607 http://arxiv.org/abs/1601.01607
[34] Justin J Miller. 2013. Graph database applications and concepts with Neo4j. In *Proceedings of the Southern Association for Information Systems Conference, Atlanta, GA, USA*, Vol. 2324. Southern Association for Information Systems, USA, 36.
[35] Jeeyun Oh, Saraswathi Bellur, and S Shyam Sundar. 2015. Clicking, assessing, immersing, and sharing an empirical model of user engagement with interactive media. *Communication Research* I, 27 (2015), 27pp. DOI: http://dx.doi.org/10.1177/0093650215600493
[36] Alexander J Quinn and Benjamin B Bederson. 2011. Human computation: a survey and taxonomy of a growing field. In *Proceedings of the SIGCHI conference on human factors in computing systems*. ACM, USA, 1403–1412.
[37] Luis FG Sarmenta. 2001. *Volunteer computing*. Ph.D. Dissertation. Massachusetts Institute of Technology.
[38] Nicu Sebe. 2010. Human-centered computing. In *Handbook of ambient intelligence and smart environments*. Springer, Germany, 349–370.
[39] Jimmy Secretan, Nicholas Beato, David B. D'Ambrosio, Adelein Rodriguez, Adam Campbell, Jeremiah T. Folsom-Kovarik, and Kenneth O. Stanley. 2011. Picbreeder: A case study in collaborative evolutionary exploration of design space. *Evol. Comput.* 19, 3 (Sept. 2011), 373–403.
[40] Takahito Seyama and Masaharu Munetomo. 2016. Development of a multi-player interactive genetic algorithm-based 3D modeling system for glasses. In *Evolutionary Computation (CEC), 2016 IEEE Congress on*. IEEE, NY, USA, 846–852.
[41] Karl Sims. 1991. Artificial evolution for computer graphics. In *Proceedings of the 18th annual conference on Computer graphics and interactive techniques (SIGGRAPH '91)*. ACM, New York, NY, USA, 319–328. DOI: http://dx.doi.org/10.1145/122718.122752
[42] Rod Sims. 1997. Interactivity: A forgotten art? *Computers in Human Behavior* 13, 2 (1997), 157–180.
[43] James M. Snell and Evan Prodromou. 2016. *Activity Streams 2.0, W3C Candidate Recommendation 15 December 2016.* Technical Report. World Wide Web Consortium. https://www.w3.org/TR/activitystreams-core/.
[44] Yu-An Sun and Christopher Dance. 2013. *Modeling Humans as Computing Resources.* Springer New York, New York, NY, 545–550. DOI: http://dx.doi.org/10.1007/978-1-4614-8806-4_41
[45] H. Takagi. 2001. Interactive evolutionary computation: fusion of the capabilities of EC optimization and human evaluation. *Proceedings of IEEE* 89, 9 (2001), 1275–1296.
[46] S.J.P. Todd and W. Latham. 1992. *Evolutionary art and computers.* Academic Press, UK. http://books.google.com.mx/books?id=KfYlAQAAIAAJ
[47] Mark Wagy and Josh Bongard. 2014. Collective design of robot locomotion. In *ALIFE 2014 Proceedings*, Vol. 14. Alife, US, 138–145.
[48] BOINC Wiki. 2012. What is volunteer computing? (2012). http://boinc.berkeley.edu/trac/wiki/VolunteerComputing [Internet; download 25-octubre-2013].
[49] Lixiu Yu and Jeffrey V Nickerson. 2011. Cooks or cobblers?: crowd creativity through combination. In *Proceedings of the SIGCHI conference on human factors in computing systems*. ACM, USA, 1393–1402.