

Fostering Creativity in the Mixed-Initiative Evolutionary Dungeon Designer

Alberto Alvarez
Malmö University
Malmö, Sweden
alberto.alvarez@mau.se

Steve Dahlskog
Malmö University
Malmö, Sweden
steve.dahlskog@mau.se

Jose Font
Malmö University
Malmö, Sweden
jose.font@mau.se

Johan Holmberg
Malmö University
Malmö, Sweden
johan.holmberg@mau.se

Chelsi Nolasco
Malmö University
Malmö, Sweden
nolascochelsi@gmail.com

Axel Österman
Malmö University
Malmö, Sweden
axelosterman@gmail.com

ABSTRACT

Mixed-initiative systems highlight the collaboration between humans and computers in fostering the generation of more interesting content in game design. In light of the ever-increasing cost of game development, providing mixed-initiative tools can not only significantly reduce the cost but also encourage more creativity amongst game designers.

The Evolutionary Dungeon Designer (EDD) [3] is a mixed-initiative tool with a focus on using evolutionary computation to procedurally generate content that adhere to game design patterns. As part of an ongoing project, feedback from a user study on EDD's capabilities as a mixed-initiative design tool pointed out the need for improvement on the tool's functionalities [4].

In this paper we present a review of the principles of the mixed-initiative model, as well as the existing approaches that implement it. The outcome of this analysis allows us to address the appointed needs for improvement by shaping a new version of EDD that we describe here. Finally, we also present the results from a user study carried out with professional game developers, in order to assess EDD's new functionalities. Results show an overall positive evaluation of the tool's intuitiveness and capabilities for empowering game developers' creative skills during the design process of dungeons for adventure games. They also allow us to identify upcoming challenges pattern-based mixed-initiative tools could benefit from.

CCS CONCEPTS

• **Theory of computation** → **Evolutionary algorithms**; • **Applied computing** → **Computer games**; • **Software and its engineering** → **Interactive games**;

KEYWORDS

Mixed-Initiative Design, Procedural Content Generation, Game Design Patterns.

ACM Reference Format:

Alberto Alvarez, Steve Dahlskog, Jose Font, Johan Holmberg, Chelsi Nolasco, and Axel Österman. 2018. Fostering Creativity in the Mixed-Initiative Evolutionary Dungeon Designer. In *Foundations of Digital Games 2018 (FDG18)*, August 7–10, 2018, Malmö, Sweden. ACM, New York, NY, USA, 8 pages. <https://doi.org/10.1145/3235765.3235815>

1 INTRODUCTION

Mixed-initiative systems in game content creation [19] refer to the combination of functions produced by procedural content generation (PCG) algorithms and human designer intentions.

In today's design paradigm, it is a common approach to have humans and machines collaborate to maximize creativity during the design process and thus software have become a backbone tool for a designer to create artifacts within the areas of architecture, consumer product and interior design. As a result, computer-aided design (CAD) has been an important facet for design practices [23]. It could be argued that game development is a fast growing application area for this facet.

Games are part of an evolving medium of creative expression, but limitations still exist in regards to its design tools' accessibility due to the fast paced life cycle and expensive nature of game development. The rising cost in game development due to games' technological evolution has resulted in a push towards automatically generated content [7, 9, 25]. Cost drivers may include multiple factors, but in the context of work processes involving human designers and artists, they are commonly identified as a huge contributor, since they are expensive. Games' complexity in design, requires the involvement of tens to hundreds of staff across a development period that may span for years. This can negatively affect a company's profitability and the development team's innovative and creative vision.

PCG approaches and functionalities are used to reduce the workload of developers, and to promote cooperation between humans and machines by providing more diverse game content that could increase quality and re-playability [15, 25]. Various development tools and level editors can be used by human designers at their disposal, making them the sole driver of the creative process.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

FDG18, August 7–10, 2018, Malmö, Sweden

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-6571-0/18/08...\$15.00

<https://doi.org/10.1145/3235765.3235815>

PCG, however, may limit the human designer’s intentions by strictly following its own algorithms, disregarding the designer’s desired parameters before generation [23]. Rather than simply being limited tools of support for the other, mixed-initiative systems can foster co-creativity in game design by combining the best of these two perspectives. Not only would it improve a development team’s overall productivity, it can also guide and improve the creativity of smaller indie teams and individuals in developing more interesting and content-rich games with less worries about development costs [15, 17].

This paper is organized as follows: Section 2 presents the related work in Mixed-Initiative design and the previous results of Evolutionary Dungeon Designer, both of them as motivation for the current work. Section 3 describes the contributions of this paper together with a description of the last release of the tool. Section 4 presents the results from the user study conducted with game developers, followed by the conclusions and future work discussed in Section 5.

2 RELATED WORK

2.1 The Main Principles of Mixed-Initiative

There are two different types of mixed-initiative [19]. The first type relates to the human creator having an idea and the computer being the mean of expression through aiding the human in their creative task (e.g. a text editor or Photoshop). The second type is described as the computer autonomously generating content and being evaluated, changed and edited by a human designer. This division is also described by Yannakakis and Togelius [25] where they present a scale with the two extremes on opposite ends: purely human design on one and purely computational design on the other. In between these extremes are varying forms applied to mixed-initiative content generation tools used for game design.

Artificial intelligence (AI) techniques have become more common and essential on aiding designers while they develop games [24]. This motivates the use of mixed-initiative systems, which promotes the co-creativity between human designers and machines providing more interesting and exciting creations [23]. However, there are still some problems when generating content, thus advocating developers still doing manual designs from scratch. Some drawbacks of completely relying on PCG is the low reliability, believability, and high predictability of the game content - all which guarantees difficulties in evaluating the generated content like, for instance, a dungeon level’s quality [12]. Therefore, by following the principles of mixed-initiative through combining the content generation with the guidance and input from a human designer, you provide aspects from both parties, hopefully limiting the weaknesses from either.

2.2 Mixed-Initiative Tools in Game Design

Recent research in the field has presented different approaches to mixed-initiative authoring tool. These are *Tanagra*, *CICERO* and the *Sentient Sketchbook*. While all these provide mixed-initiative interfaces to the designer, they also share the limitation of addressing a specific game type.

Sentient Sketchbook aims at generating maps for strategy games such as *Starcraft* [16]. Users can sketch a low-resolution map that

seeds an underlying evolutionary algorithm that provides suggestions. Low-resolution sketches reduce the creative strain on the user during the design process, but also makes it easier for the program to detect patterns in the map. Once the user deems the generated and edited low-resolution map good enough, the program can then generate the higher resolution map while still maintaining the patterns that were detected in the sketch.

Tanagra is used to develop 2D platformer levels [20], while still checking whether the generated content is playable or not. *Tanagra* offers users an empty grid where they can place different tiles such as floor, enemies, and coins. Mixed-initiative is implemented so that users can select tiles and objects they want to keep in their designs, while *Tanagra* generates new content around them.

CICERO focuses on the generation of dungeons for adventure games [17]. *CICERO* offers users the possibility to define the behavior of the game components they include in their designs, such as power-ups, win and lose conditions, and collision-triggered behavior. From these definitions *CICERO* recommends different game mechanics that would suit the game, such as the optimal weapon types to include in the game. By manually editing game content in the dungeon and having *CICERO* run the game and test different element combinations, users can understand how different layouts affect the generated gameplay.

2.3 Dungeon Design in Videogames

The dungeon is a popular level design archetype found in several popular game genres [5, 18]. Dungeons are also popular in PCG research, where different approaches have been presented for generating dungeon levels [8, 11, 13, 19, 22]. These works emphasize the importance of considering goals, missions, the narrative or themes, visual style, and gameplay rules when designing levels, therefore they should be taken into consideration when developing a mixed-initiative tool for content generation [12]. These factors are mostly decided by the human designer, thus a designer has to be integrated into the dungeon generation process.

Another key aspect to dungeon generation is player progression [6]. Designers ensure that the player’s experience throughout a level will be coherent and effective, which will be affected by the content they create. This includes reward and challenge balancing among the rooms in a dungeon.

2.4 The Evolutionary Dungeon Designer

Previous research presented the *Evolutionary Dungeon Designer* (EDD) [3, 4] as a mixed-initiative authoring tool for designing dungeon rooms for adventure games. EDD automatically generates and suggests rooms to the user while the user is manually designing one of them. The user either form the room from scratch or from a previously generated suggestion. This is done by means of a FI-2Pop GA [14], where game design patterns are used both as input parameters and as objectives. These patterns involve micro-patterns (ENEMY, TREASURE, CHAMBER, CORRIDOR, CONNECTOR, ENTRANCE, and DOOR) as well as meso-patterns (AMBUSH, GUARD CHAMBER, TREASURE CHAMBER, and GUARDED TREASURE). EDD also ensures that all generated rooms are playable.

Initial experiments on EDD [3] validated its PCG system in terms of fitness optimization, pattern detection, and solution diversity,

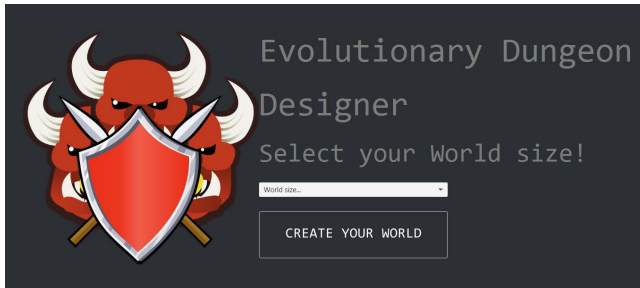


Figure 1: The start screen lets users choose the dungeon dimensions.

providing a sufficient level of control to the designer. The following iteration [4] explored the capabilities of EDD as a mixed-initiative level generator as a means of facilitating collaboration between human designers and PCG algorithms. Among its key features, the participants of a user study highlighted EDD as a useful framework for working with game design patterns in the context of search-based problems. The suggestions were considered a good source of inspiration as well as time saving. This user study also shaped the roadmap for future improvements on EDD. This included extending EDD from room generation to complete dungeon generation, and preserving the users' designs to a higher degree in relation to both design patterns and room aesthetics.

This version of EDD extends previous work based on the aforementioned user study by implementing the following key improvements:

- The designer is now able to construct, develop, and edit a grid-based dungeon of different dimensions and interconnected rooms, in contrast to a single room, which in turn, helps the designer on having context over their work on individual rooms and giving them more freedom on producing variations.
- The designer receives extended information about the consequences of their changes in individual rooms, and the differences between the current edited room and the proposed suggestions by the EA.
- The UI has been renovated to account for the newly added features by means of different views and options, as well as, a better structure and distribution of the different elements in the generator.
- Navigation tools have been added within a view and between views, which provides an overview of the dungeon, along with a better context of the edited room.
- The EA has been updated to assess and preserve the aesthetic criteria of the designer by means of a new capability of locking sections of an edited room for preserving custom aesthetic structures, and by extending the evaluation function through the measurement of symmetry and similarity in the provided suggestions, both which are further explained in [2].

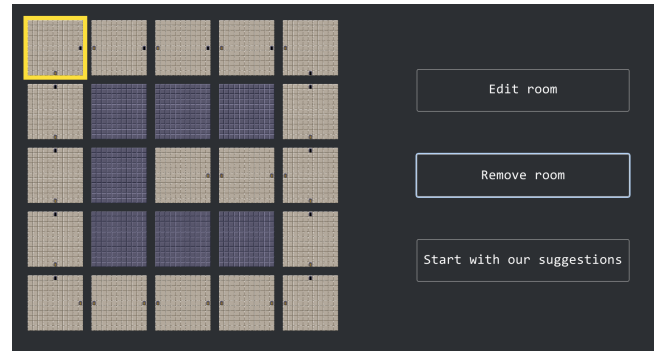


Figure 2: Sample world view showing a 5x5 dungeon with 7 disabled rooms. User actions are displayed in the rightmost buttons.

3 IMPROVING THE MIXED-INITIATIVE EVOLUTIONARY DUNGEON DESIGNER

Figure 1 shows the start screen in EDD, which starts a new workflow by prompting users to choose the maximum number of rooms in the dungeon to be developed. The dimensions range from 2x2 rooms up to 7x7 rooms in a square dungeon grid (also referred as world grid). From this point, the workflow offers users three different views: 1) a world view for dealing with aspects regarding the dungeon as a whole; 2) a room view which places the focus in a particular room in the dungeon; and 3) the suggestions view, which produces six different suggestions with diverging room configurations (e.g. more corridors or more chambers) for the user to choose from. The user can freely alternate between views during the design process. The current dungeon layout can be saved at any moment from either of the views.

The world view (Figure 2) opens up right after the start screen, displaying a grid of the selected size composed by a fully connected set of empty rooms (all rooms are connected to their neighbors). The users can load a previously saved dungeon design, skipping the start screen and resume work from the state in which the dungeon design was saved.

From the world view users can then click and select any room to:

- disable or enable the room. Disabling makes the room inaccessible, removing all doors from the adjacent rooms. This can be undone by clicking *enable*. Single rooms that become isolated after all their neighbors have been disabled, are automatically disabled as well. Figures 3a and 3b show two examples of dungeons with several disabled rooms,
- get procedurally generated suggestions for that room in the suggestions view,
- load the room in the room view for manual editing.

3.1 The Suggestions View

By selecting “Start with our suggestions” in the world view (Figure 2) six uniquely generated rooms are presented to the user in a separate window: the suggestions view (Figure 4). When clicking

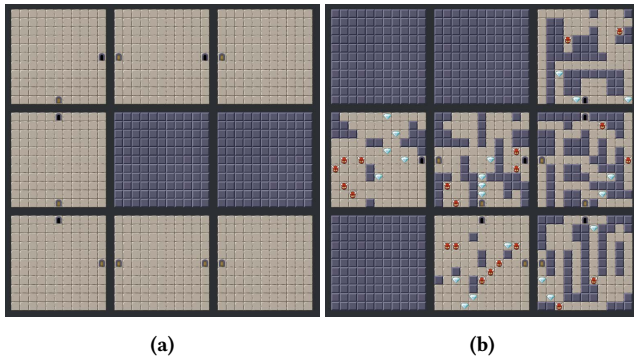


Figure 3: Sample (a) 3x3 dungeon with 2 disabled rooms and 7 empty rooms, and (b) 3x3 dungeon completed dungeon with 3 disabled rooms.

any of the suggestions, it will replace the previously selected room in the dungeon.

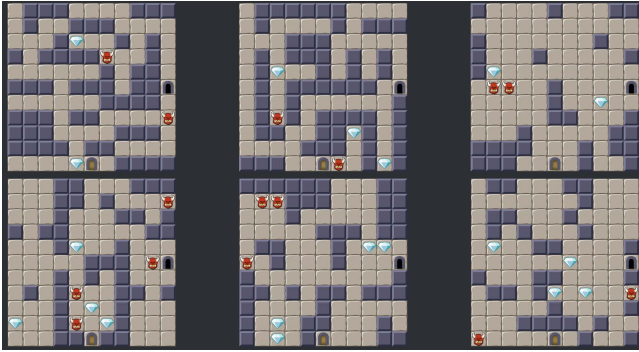


Figure 4: Six procedurally generated rooms presented to the user in the suggestions view.

A similar functionality was present in the former version of the tool, presented only once as the start screen. Now users can freely alternate between the world and the suggestions views, getting as many suggestions as they need, deciding whether to start creating every room from a clean state or to get inspiration from one of the generated rooms.

Suggestions preserve the door layout from the room that was selected in the world view. The suggestions shown in Figure 4 have been created for the room selected in Figure 2, placed in the top-left corner, and containing only two doors connecting them to their east and south neighbor.

3.2 The Room View

Users edit single rooms in the room view (Figure 5), regardless of whether these are new empty rooms, procedurally generated suggestions, or previously edited rooms. The room view is an improved and extended version of the main screen in the last version of EDD [4]. All functionalities from that version are still present: manually editing the room by changing tiles (floor, wall, enemy,

and treasure), displaying an overlay view of the existing design patterns, and procedurally generating suggestions based on the current edited room's configuration.

Navigation is one of the crucial added features, allowing users to move around the dungeon without going back to the world view. Two other options offer navigation through the dungeon: the navigational buttons and the minimap. The navigational buttons are displayed next to each of the edited rooms' borders that contain a door. Provided that the room being edited in Figure 5 is located in the top-left corner, two navigational buttons are displayed right and below the room, respectively. Clicking a navigational button transports the user to that room, replacing the currently edited room with the targeted neighbor. Instead of using arrows or any other fixed picture, these buttons preview the neighboring rooms as a hint for users to help them design the room currently being edited. The navigational buttons are automatically refreshed to reflect up-to-date changes performed to the neighboring rooms.

The minimap displays a scaled-down overall picture of the whole dungeon, highlighting the currently edited room with a yellow border. Users can navigate to any room, which is not disabled and is displayed on the minimap by clicking on it, replacing the current room. The buttons above the minimap allow users to go *Back To World Grid*, to *Update Minimap*, as well as request and select procedurally generated suggestions. The whole minimap is updated whenever the user navigates to a different room, but if the user wants to see the last changes applied to currently edited room reflected on the minimap, a manual refresh has to be done. This is done to reduce the workload derived from re-rendering the minimap automatically after every manual edition.

The generated suggestions work similarly to the previous version of EDD: four unique maps are generated by the underlying evolutionary algorithm in four subsequent evolutions, seeding the four initial populations with different sets of features extracted from the edited room. Each suggestion is evolved by means of a different fitness function, therefore addressing different goals to maximize diversity in the provided suggestions. Clicking on a suggestion highlights it, and clicking *Apply Suggestion* replaces the current room with the highlighted map. This differs from the previous version, in which maps were applied at the moment they were clicked on, occasionally causing work loss due to accidental replacements.

Additionally, highlighted suggestions display informative parameters below them. These describe meaningful features of the highlighted room that are relevant to both the human designer and the evolutionary algorithm's fitness calculation: number of enemies and treasures, enemy and treasure rate (in relation to floor tiles), and entrance and treasure safety (see [3] for a detailed description). These parameters are displayed as a comparison between their values in the edited room and in the highlighted suggestion, showing how they would change if the suggestion is applied.

Two checkboxes below the suggestions now offer users the possibility to ask specifically for the provided suggestions to address symmetry and similarity aesthetic features, respectively. By ticking the symmetry checkbox, two of the suggestions will be generated by the evolutionary algorithm using a symmetry fitness function, which enable the generation of symmetric rooms, (either vertically, horizontally, or diagonally). Analogously, ticking the similarity

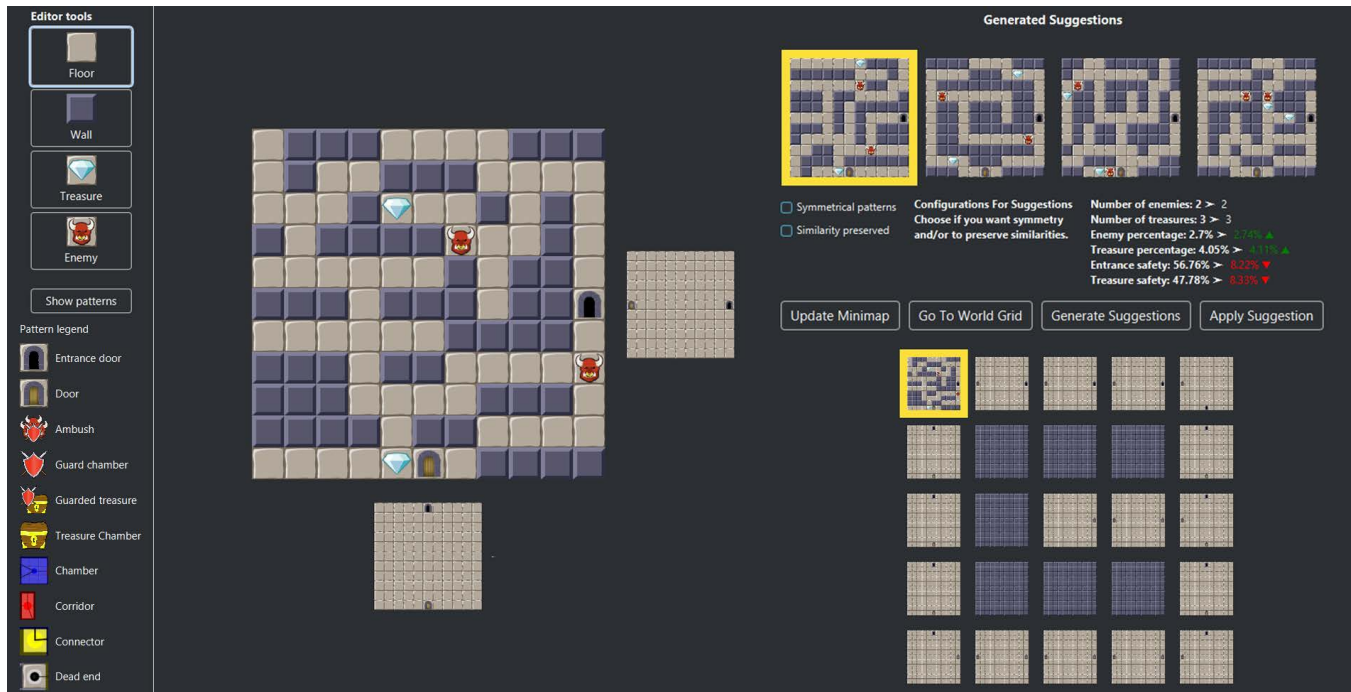


Figure 5: The room view while editing the top-left corner in a 5x5 dungeon with seven disabled rooms.

checkbox, the other two suggestions are generated with a similarity fitness function, which promotes the generation of room aesthetically similar (but never equal) to the currently edited map. These fitness functions are presented in [2]. When both checkboxes are unchecked, the fitness functions described in [4] are used.

4 USER STUDY

A user study was conducted in order to assess the impact on the design process caused by the improvements made to EDD. Five game developers participated in the study, which had the following structure:

- *Introduction to the purpose of the study.* Participants were asked whether they were familiar with the previous version of EDD.
- *Demonstration of the tool,* showcasing its workflow and features with a short example performed over a 3x3 dungeon.
- *Designing a dungeon.* After the demonstration the users were tasked to design a 3x3 dungeon within approximately 10 minutes, saving the work after that for a later analysis and discussion conducted in a structured interview with the participant after this phase. Two observers took notes of what the participant was doing, providing additional data for the later analysis.
- *Questionnaire.* The users were asked a few questions regarding their background in game design as well as dungeon-based games. They were also asked whether they had any previous experience with mixed-initiative tools.

- *Interview.* A semi-structured interview was conducted to provide data for an analysis and discussion about the tool, and its improvements. Audio was recorded for a later analysis.

As a result of the questionnaire, the following information was gathered from the participants:

- User 1 has been working for more than ten years in the game industry as a data scientist and user experience researcher. The user holds prior experience with RPGs and dungeon crawlers, and is familiar with the terms of mixed-initiative tools and has used The Sentient Sketchbook in the past. This user is the only one who participated in the former user study of EDD.
- User 2 has been working for six months as a project coordinator of eSports events and has long experience of playing dungeon crawlers and RPGs. This user is not familiar with mixed-initiative concepts and has never used a mixed-initiative authoring tool before.
- User 3 has been working for six years in the game industry as a user experience researcher and a biometrics expert. The user has prior experience with dungeon style games, but has limited knowledge about mixed-initiative tools.
- User 4 has been working for nine years as a senior user experience researcher and has long experience of playing dungeon crawlers and RPGs. This user is not familiar with mixed-initiative concepts and has never used a mixed-initiative authoring tool before.
- User 5 has been working for three weeks as a game user researcher. This user has no experience with dungeon crawlers

Table 1: General consensus on EDD’s features

Description	Participants’ Consensus
World Grid of the dungeon	Its purpose of establishing an illusion of a fully realized dungeon is somewhat achieved. However, limitations exist with how it defines feasibility, a dungeon’s starting point, and the entrances, which disrupts the designers’ decisions.
World View	The world view’s usefulness for the most part could not be established, other than for the purpose of going to the suggestions view (which was already seldom during the user study) and having a closer look at the entire dungeon without any distractions. Some participants preferred features to be already in the room view’s minimap, and some wanted to see more specific functionalities within the world view itself.
Enabling and disabling rooms	As the user study restricted participants to create 3x3 dungeons, this feature for the most part has been neglected. This is also in part because of its accessibility only being in the world view, which proved to be an inefficient view in general. However, its use for bigger dungeon sizes later on was appreciated, especially for more intricate design purposes.
Suggestions View	Similarly to enabling and disabling rooms, it was quite difficult to encourage the use of this functionality due to the world view’s inefficient usability. However, this could also be due to the dungeon’s small size, as some participants expressed high interest in using more suggestions with larger dungeon sizes.
Minimap and navigation	The minimap proved to be a strong tool not only for navigation purposes, but also for supporting design decisions and choices. The directional buttons were rarely used, but their room previews were helpful in emphasizing the current room’s connection to adjacent rooms without looking at the minimap. On the other hand, this lowered the usability of the world view.
Parameters	The parameters were, in general, lacking. They served to be important in decision-making when choosing a suggested map in room view, but there were still doubts on their accuracy and sufficiency when providing information about the generated suggestions.
Generated maps for suggestions in room view	Suggestions in the room view proved to be very helpful in supporting the whole design process as they primarily acted as inspirations for the users. The most prominent comment among the users is the preference of having more control on how suggestions should be generated depending on different types of parameters.
Design patterns	The patterns’ visualization was, in general, lacking and not self-explanatory. Some participants have expressed interest in using patterns as a parameter in the generation of suggestions.
Dark theme	EDD’s dark theme for the user interface received a positive response as it makes working with the program easier.

and dungeon-based RPGs, and is not familiar with mixed-initiative tools.

5 RESULTS AND DISCUSSION

As with any raw qualitative data, the data collected from the user study has to go through a condensation process in order to isolate the most relevant information that will answer the research questions. In a blueprint provided by Srnka and Koeszegi [21], the qualitative data obtained has undergone four out of five stages:

- *Material sourcing*: an audio recording the user study, interview materials, and the authors’ own observations.
- *Transcription*: combining and writing down the observations and questionnaire answers for each participant in the user study.
- *Unitization*: dividing the data according to the mixed-initiative features of EDD.
- *Categorization*: dividing the data according to categories relevant to the research questions while taking into consideration the principles of mixed-initiative.

All participants in the user study perceived EDD as overall good and intuitive. Table 1 shows their general consensus of EDD’s usability and capability to foster creativity in dungeon design. Table 2 lists the participants’ most requested missing features.

The main goal of mixed-initiative interaction pertains to the flexibility of roles between the human and computer as a team and simplifying the general experience [1], and this was somewhat achieved by EDD. This could be proven by how features such as suggestions and the implementation of a whole dungeon with navigation have definitely supported the users when making decisions throughout the design process. As a result the experience was overall simple and intuitive. It could not be said, however, that the set goal has been fully achieved; a fully successful mixed-initiative system emphasizes interchangeable roles of the human and computer while maintaining the balance between them. The participants in the user study did not feel restricted, but they still desired more control in EDD’s assistance in the design process, as well as different suggestions that the designer cannot come up with themselves.

Horvitz [10] provides a list of principles for mixed-initiative user interfaces which would enhance human-computer interaction. EDD has achieved four out of twelve in Horvitz’s list of critical factors that would make up a fully successful mixed-initiative system:

Table 2: Participants’ most requested features

Feature	Description
Design patterns	Their visualization and accuracy should be improved. Other than acting as visual guide for map information, they should be used to help generate rooms as well. They should also be available for the entire dungeon.
Parameters	They need to have more information about the specific room, and have better visualization in order to make the designer trust their accuracy more. The parameters should also consider the entire dungeon as a whole in different terms such as difficulty and balance.
Generated suggestions	In general, the participants want more variety and control in the generation of suggestions using different types of parameters e.g. their degree of similarity and fitness functions.
Redefined feasibility	Eddy 3.0’s definition of feasibility should be revised which considers the whole dungeon and its connected rooms.
World View	The World View should be revised and enhanced with more special features which would encourage users to visit it more.
World grid	The computation of the whole dungeon should be improved. It should have an option to define a starting point. Its definition of entrance doors should be improved, as well as the calculation of distances of tile types.
Version control and previews	Some participants want to preview suggestions within the Room View to help their judgment and the ability to save suggestions for later use. They also want to revert to old designs in case they have second thoughts.
Templates	Some participants want the ability to save their own manual designs to be carried over to other grids.
Automated assistance	The participants in general welcome a bit more automated assistance when doing manual designs, which can reduce clicking around the program. It should also not be too invasive for the designer.

- *Developing significant value-added automation*: providing an automated solution that cannot be achieved with direct manipulation. EDD provides a framework for the generation of complex dungeons of different sizes, together with suggestions of similar dungeon rooms and information parameters for these suggestions.
- *Considering uncertainty about a user’s goals*: taking advantage of a user’s uncertainty in their intentions. EDD provides the choice to initialize rooms in a dungeon with either an empty slate or from any of the generated suggestions.
- *Inferring ideal action in light of costs, benefits, and uncertainties*: considering the value of an automated service in regards to the usually expected value of taking actions. EDD’s main motivation is to significantly reduce the cost of game design while maintaining and improving creativity, which has at least partially been fulfilled.
- *Employing dialogue to resolve key uncertainties*: establishing an efficient dialog between the human and computer when uncertainty arises while considering the costs of potentially disrupting the user. EDD extracts and displays relevant features in the edited and suggested rooms for the users to guide their decisions. The minimap also fulfill parts of this role.

There are other principles which are relevant to EDD which fall in line with the participants’ feedback. For example, some principles such as the ability to continuously learn from the user’s input and to preserve memory of their decisions and actions may pertain to the desired features of having more control in the generation maps and receiving more assistance in preserving their own manual designs for different purposes.

6 CONCLUSIONS AND FUTURE WORK

The contributions presented in this work explore how the user interface and the mixed-initiative aspects in the Evolutionary Dungeon Designer have been improved, as well as how they should be improved on further in order to increase creativity during the dungeon design process.

The addition of the world grid provides the adoption of a new workflow, which offers users the possibility to start designing either from empty rooms or PCG suggestions. Various changes in the user interface were made to accommodate the increased dungeon size. With a larger dungeon, navigation has proved to be a key functionality, as well as giving an overview of the adjacent rooms. Now users can get a better understanding of the context of the room currently being edited. In conjunction with the navigation and larger dungeons, a minimap was also added to further enhance the experience when designing a larger dungeon. Alongside these changes, aesthetic goals have been included in the generative process. Visual cues for room descriptors were added, so that the user can make a more informed decision when selecting suggested maps.

Compared to its previous version, EDD further empowers the mixed-initiative design process by providing more context, feedback, flexibility, and to some extent, the ability to address aesthetic features in the procedural suggestions. Creativity can directly adhere to the amount of interesting possibilities a designer can employ, which is relevant to providing rich contexts to dungeon designs. EDD offers a mixed-initiative experience that provides adequate flexibility for the designer’s intentions as the results from the user study have shown.

Overall, our user study successfully shows the strengths of mixed-initiative tools for designers but it also reveals various limitations,

which should be considered by the community when creating a mixed-initiative tool.

To a certain extent, controllability is preferred than expressivity, as the users continuously try to impose their vision, which is a non-trivial task for automated systems to capture, thus, the users are more likely to sacrifice to a certain degree expressivity and exploration of the tool by gaining control over the generated content.

The capability of proposing useful and novel suggestions is fundamental to fostering creativity and impulses the generation of more interesting content. Moreover, explicit information about the designers' changes and choices is important as it helps them understand the effect of their decisions.

Finally, this work has identified features that should still be taken into consideration for future versions of the tool, which are shown in Table 2.

ACKNOWLEDGMENTS

The Evolutionary Dungeon Designer is part of the project *The Evolutionary World Designer* which is supported by The Crafoord Foundation.

REFERENCES

- [1] JE Allen, Curry I Guinn, and E Horvitz. 1999. Mixed-initiative interaction. *IEEE Intelligent Systems and their Applications* 14, 5 (1999), 14–23.
- [2] A. Alvarez, S. Dahlskog, J. M. Font, J. Holmberg, and Johansson. S. 2018. Assessing Aesthetic Criteria in the Evolutionary Dungeon Designer. In *Proceedings of the 13th International Conference on the Foundations of Digital Games (FDG '18)*. ACM, New York, NY, USA.
- [3] A. Baldwin, S. Dahlskog, J. M. Font, and J. Holmberg. 2017. Mixed-initiative procedural generation of dungeons using game design patterns. In *2017 IEEE Conference on Computational Intelligence and Games (CIG)*. 25–32. <https://doi.org/10.1109/CIG.2017.8080411>
- [4] A. Baldwin, S. Dahlskog, J. M. Font, and J. Holmberg. 2017. Towards Pattern-based Mixed-initiative Dungeon Generation. In *Proceedings of the 12th International Conference on the Foundations of Digital Games (FDG '17)*. ACM, New York, NY, USA, Article 74, 74:1–74:10 pages.
- [5] Blizzard North. 1996. *Diablo*. (1996).
- [6] Eric Butler, Adam M Smith, Yun-En Liu, and Zoran Popovic. 2013. A mixed-initiative tool for designing level progressions in games. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*. ACM, 377–386.
- [7] Liam Doherty, Max Whitney, Jurika Shaky, Mayo Jordanov, Patrick Loughheed, David Brokenshire, Shilpi Rao, Samir Menon, and Vive Kumar. 2005. Mixed-initiative in computer games: Algorithmic content creation in open-ended worlds. In *AAAI Fall Symposium on Mixed-Initiative Problem-Solving Assistants*. aaai.org, 46–56.
- [8] J. Dormans. 2010. Adventures in Level Design: Generating Missions and Spaces for Action Adventure Games. In *Proceedings of the 2010 Workshop on Procedural Content Generation in Games*. ACM, New York, NY, USA.
- [9] J. M. Font, R. Izquierdo, D. Manrique, and J. Togelius. 2016. Constrained Level Generation Through Grammar-Based Evolutionary Algorithms. In *Applications of Evolutionary Computation: 19th European Conference, EvoApplications 2016, Porto, Portugal, March 30 – April 1, 2016, Proceedings, Part I*, G. Squillero and P. Burelli (Eds.). Springer International Publishing, Cham, 558–573.
- [10] Eric Horvitz. 1999. Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. ACM, 159–166.
- [11] L. Johnson, G. N. Yannakakis, and J. Togelius. 2010. Cellular Automata for Real-time Generation of Infinite Cave Levels. In *Proceedings of the 2010 Workshop on Procedural Content Generation in Games*. ACM, New York, NY, USA.
- [12] D. Karavolos, A. Bouwer, and R. Bidarra. 2015. Mixed-Initiative Design of Game Levels: Integrating Mission and Space into Level Generation. In *Proceedings of the 10th International Conference on the Foundations of Digital Games*.
- [13] D Karavolos, A. Liapis, and G. N. Yannakakis. 2016. Evolving Missions to Create Game Spaces. In *Proceedings of the IEEE Conference on Computational Intelligence and Games*.
- [14] S. O. Kimbrough, G. J. Koehler, M. Lu, and D. H. Wood. 2008. On a Feasible-Infeasible Two-Population (FI-2Pop) genetic algorithm for constrained optimization: Distance tracing and no free lunch. *European Journal of Operational Research* 190, 2 (2008), 310 – 327.
- [15] A. Liapis, G. N. Yannakakis, and J. Togelius. 2013. Generating Map Sketches for Strategy Games. In *Proceedings of Applications of Evolutionary Computation*, Vol. 7835, LNCS. Springer.
- [16] A. Liapis, G. N. Yannakakis, and J. Togelius. 2013. Sentient Sketchbook: Computer-Aided Game Level Authoring. In *Proceedings of the 8th Conference on the Foundations of Digital Games*.
- [17] Tiago Machado, Andy Nealen, and Julian Togelius. 2017. SeekWhence a Retrospective Analysis Tool for General Game Design. In *Proceedings of the 12th International Conference on the Foundations of Digital Games (FDG '17)*. ACM, New York, NY, USA, Article 4, 6 pages. <https://doi.org/10.1145/3102071.3102090>
- [18] Nintendo Research & Development 4. 1986. *The Legend of Zelda*. (1986).
- [19] N. Shaker, J. Togelius, and M. J. Nelson. 2016. *Procedural Content Generation in Games: A Textbook and an Overview of Current Research*. Springer.
- [20] G. Smith, J. Whitehead, and M. Mateas. 2011. Tanagra: Reactive Planning and Constraint Solving for Mixed-Initiative Level Design. *IEEE Transactions on Computational Intelligence and AI in Games* 3, 3 (Sept 2011), 201–215.
- [21] Katharina J Srnka and Sabine T Koeszegi. 2007. From words to numbers: how to transform qualitative data into meaningful quantitative results. *Schmalenbach Business Review* 59, 1 (2007), 29–57.
- [22] R. van der Linden, R. Lopes, and R. Bidarra. 2013. Designing procedurally generated levels. In *Proceedings of the 2nd AIIDE Workshop on Artificial Intelligence in the Game Design Process*.
- [23] G. N. Yannakakis, A. Liapis, and C. Alexopoulos. 2014. Mixed-Initiative Co-Creativity. In *Proceedings of the 9th Conference on the Foundations of Digital Games*.
- [24] G. N. Yannakakis and J. Togelius. 2015. A Panorama of Artificial and Computational Intelligence in Games. *IEEE Transactions on Computational Intelligence and AI in Games* 7, 4 (Dec 2015), 317–335. <https://doi.org/10.1109/TCIAIG.2014.2339221>
- [25] Georgios N. Yannakakis and Julian Togelius. 2018. *Artificial Intelligence and Games*. Springer. <http://gameaibook.org>.