``What is a Super Mario level anyway?" An Argument For Non-Formalist Level Generation in Super Mario Bros.

Tommy Thompson
Anglia Ruskin University
Cambridge, UK
tommy@t2thompson.com

ABSTRACT

The video game series *Super Mario Bros.* has proven immensely popular in the field of artificial intelligence research within the last 10 years. Procedural content generation research in *Super Mario* continues to prove popular to this day. However much of this work is based largely on the notions of creating 'Mario-like' level designs, patterns or structures. In this paper, we argue for the need to diversify the generative systems used for level creation within the *Super Mario* domain through the introduction of more aesthetic-driven and 'non-formalist' approaches towards game design. We assess the need for a broader approach to automated design of Mario artefacts and gameplay structures within the context of *Super Mario Maker*: a *Super Mario Bros.* level creation tool. By assessing a number of top-ranked levels established within the player community, we recognise a populist movement for more radical level design that the AI community should seek to embrace.

Keywords

Procedural content generation, level design, Super Mario Bros.

INTRODUCTION

Automated game design (AGD) is a field of research and development that focusses on the creation of entire games: ranging from traditional card and board games to that of video games. This is a immensely tasking yet creative field, with a large body of research typically found within the procedural content generation (PCG) community: where emphasis is placed on creating specific assets such as the interactive space (levels/maps), weaponry, quests and story-lines. The contributions of this work has helped to nurture a slow and continual growth in research focussed on the algorithmic creation of games in their entirety.

One of the larger challenges faced when working in AGD is establishing the rules that comprise the core logic of the gameplay experience. Much of the research in AGD adopts the approach of establishing rule-sets courtesy of a search-based generation procedure, in which new rules can be formally established, broken down and recombined in ways that typically human designers would not think of doing. It is through this iterative process that 'interesting' gameplay experiences can emerge. This process and its subsequent output raises questions of the nature of games and play: is this generated content still technically a game if it deviates from the structures previously established? Does this merit inclusion within the existing taxonomies of games and on what basis? This notion of whether a game is in fact

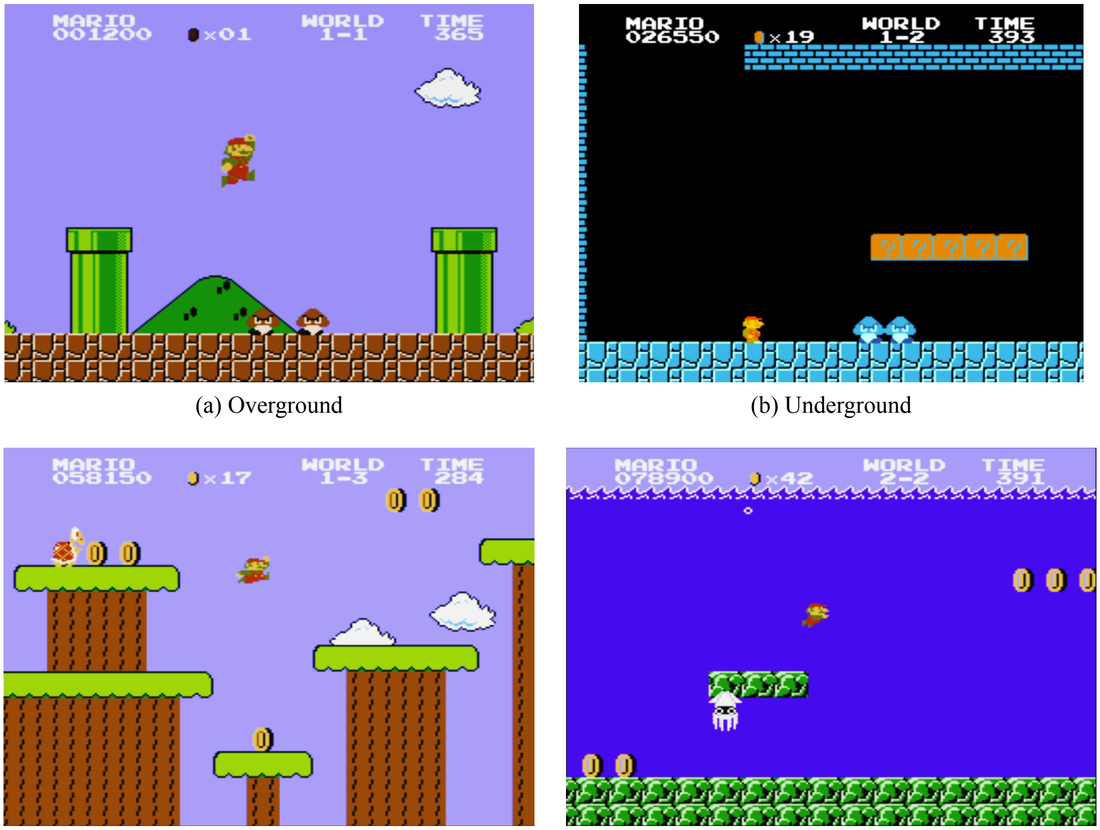
Proceedings of DiGRA 2016: DeFragging Game Studies

‘a game’ has been a focus of discussion in most recent years within the video game community. Recent titles such as *Gone Home* (The Fullbright Company, 2013), *Proteus* (Key & Kanaga, 2013) and *Everyone’s Gone to the Rapture* (The Chinese Room, 2015) have been labelled ‘non-games’ and derided by some circles of the gaming community based on conservative expectations of characters, interactions, settings and themes. In each instance, these games deviate from traditional modes of interaction and narration with an emphasis on more aesthetic qualities. In the case of *Gone Home* and *Everyone’s Gone to the Rapture*, emphasis is placed on non-linear storytelling through the use of an interactive environment. Meanwhile *Proteus* is devoid of any purpose, agenda or narrative: with players encouraged simply to explore the procedurally generated world. In each case, rules of interaction are established as a framework such that players may experience the qualities hidden within the game world. This approach deviates significantly from ‘traditional’ game design, in which modes of interaction are established such that players may progress towards a prescribed goal. Said goals are achieved within the game through an element of skill: utilising the interactions permitted to overcome challenges presented before the player. Given these tropes have long been established throughout the history of the video gaming genre, games of this nature receive criticism from sections of the community as a result of their non-conformance. This is to be expected, given that the majority of high-profile video game releases, their content and design are driven by the predilections of a mainstream market which the games industry[1] itself nurtures.

A recent argument detailed in [2] highlighted the need for automated game design to deviate from this “formalist” view of game design that is largely the source of this conservative backlash. While there should be an effort to ensure that rules, mechanics and dynamics are established in generated games, the authors argue for greater emphasis to be placed on “non-formalist” areas of design: focussing on aesthetic and the potential to illicit an emotional response from the player. Much of this argument stems from a need to address these larger industry-trends at a time when AGD is still in its relative infancy. By formalising a non-formalist approach at an early stage, it ensures not only that the creative potential of the AGD field is not stifled but also that designers place a greater emphasis on the (often implicit) biases injected into systems they have developed.

This paper aims to support the arguments raised in [2] by demonstrating the potential for this approach to game design. This is achieved by raising the importance of a particular problem domain that will be familiar to the AGD and more specifically the PCG community: the *Super Mario Bros.* series. This series of two-dimensional (2D) platformers is typified by the inaugural release of *Super Mario Bros.* (Nintendo R&D4, 1985) in which players take control of the Mario avatar with the task of navigating an environment filled with hazards and traps. This game and the subsequent body of titles released over a period of 30 years have broadened and reinforced the core rules, objectives, obstacles and dynamics expected of the series and indeed the 2D platforming genre. While each game allows for a broad range of gameplay opportunities to be afforded of the player, they are still highly constrained and in many respects help reinforce a conservative approach towards game design. This formalist aspect has subsequently been transposed, albeit implicitly, upon research in *Super Mario Bros.* level design, with the majority of research in this field focussed on level structure and

[1] In this instance referring to what is commonly identified as the ‘AAA games industry’.



(a) Overground

(b) Underground

(c) High-ground

(d) Underwater

Figure 1: Screenshots of *Super Mario Bros.* in a variety of locales.

rules implied from releases within the series [11, 8, 19]. This has in many respects silently enforced the notion that 'validity' of generated content in this problem space is dictated solely by 'playability' rather than any non-traditional aesthetic qualities.

It is with this in mind we re-frame the conservative arguments raised in this introduction for our purpose: *'What is a Super Mario level?'* We argue that the non-formalist model of aesthetics and play discussed in [2] can be adopted in research based around *Super Mario Bros.* We examine existing PCG research within the *Super Mario Bros.* domain and how the formalist model of Mario design has been reinforced. We subsequently examine how the established notions of the *Super Mario Bros.* series itself are challenged by the content created by the player-community of Nintendo's *Super Mario Maker* (Nintendo EAD, 2015) and openly promote non-formalist approaches to game design. We identify numerous forms in which this content takes shape and how more aesthetic-driven PCG could be explored within this otherwise formalist system.

THE SUPER MARIO BROS. SERIES

Super Mario Bros. (SMB) is the first in a series of 2D platforming games that entrusts players with the task of navigating one of the two titular brothers, Mario or Luigi[2] through the hazards of the Mushroom Kingdom in an effort to rescue Princess Toadstool (Peach) from the antagonist Bowser. To achieve this players must walk, run and jump across environments comprised of platforms, gaps, obstacles, traps and enemies. In addition, players are expected to collect coins and utilise power-ups that provide either increased health or new gameplay mechanics. The core mechanic of the *Super Mario* series is the ability to jump: which can be used to kill enemies, avoid projectiles and in later releases navigate high-structures or impassable terrain courtesy of wall jumps and spin-jumps respectively.

Gameplay is segregated into 'levels' of play: in which players must navigate from the starting point to an obstacle at the far right of the map, typically a flagpole. The clearance of this obstacle while non-negotiable is easily achieved, with a score attributed to its completion to encourage optimal play. This is a significant deviation from the original *Mario Bros.* (Nintendo R&D1, 1983) in that players must continue to explore the world by scrolling the view-frame along the x-axis. The locale of levels varies throughout play, with locations on land, underground, on high ground or underwater (Figures 1a through 1d respectfully). In each instance, the locale may impose restrictions on movement through the environment: with underwater levels imposing a fresh set of controls and gameplay mechanics. These levels are often grouped together to form 'worlds' that embrace specific game design concepts or art styles for a fixed period of the game. Worlds conclude with the the final level typically being a fortress or castle, where a larger enemy must be defeated in combat.

A Reduction of *Super Mario Bros.*

Having identified some of the core mechanics of the *Super Mario* series, we now identify a reduction of the *Super Mario* level structure such that we make clear to the reader the range of creative flexibility that is available to designers. This allows us to establish the space that levels created either as part of *Super Mario* titles or by academic research can be found within.

We first consider the functional aspects of the games structure; looking at the mechanics, dynamics and aesthetic qualities of traditional *Super Mario* games. To do so we adopt the Mechanics, Dynamics and Aesthetics (MDA) framework detailed in [7] as means of understanding *Super Mario* gameplay. This allows us to establish the core elements of the gameplay experience. The completion reduction is detailed in Table 1.

As noted in Table 1, the player-driven dynamics of *Super Mario Bros.* are largely reliant upon the collision mechanic being adopted repeatedly: with a strong emphasis on chaining multiple collisions together to form a specific end-result. These dynamics are suitably exploited in the aesthetic elements that drive the 'fun factor' of *Super Mario* games, with a focus notably on challenge and sensation. While there are a range of aesthetic qualities to be found in *Super Mario* games, there are still opportunities to be found within this gameplay space given it fails to satisfy all aesthetics established within the MDA framework[3].

[2] In *Super Mario Bros.*, Luigi was the avatar provided for the second player in the event of a two-player game. The avatar itself simply a palette-swap of the Mario avatar. However he has since became a unique

Table 1: Reduction of *Super Mario* games through use of the MDA framework [7].

MECHANICS		
Movement	Avatar moves along the x- and y-axis. The former is achieved by moving a fixed distance per frame when in a walking or running state. Y-axis movement achieved by a jumping mechanic restricted by gravitational forces.	
Collision	Avatar may collide with objects, with resulting effect determined by orientation of collision and state of the colliding object.	
	Blocks	Blocks do no harm to the player, but the player can strike them by hitting them from underneath (or in the case of future *Super Mario* titles, slamming onto them from above). This can result in the destruction of this block or the changing of its state.
	NPCs	When colliding with non-player characters (NPCs) and projectiles, the player avatar is vulnerable on its top, left and right sides. Jumping atop a NPC will typically result in its death, unless it is designed to counter this attack (e.g. The Spiny enemy wears a shell of spikes on its back).
	Items	Power-ups such as the 'Super Mushroom' and 'Fire Flower' will trigger a change in player health, abilities and/or sprite in the event of collision.
Terminal States	Negative terminal state achieved should avatar exceed lowest y-value of the gameplay space or collide with an enemy while in a basic/weakened state. Positive terminal state achieved in the event avatar collides with end-level obstacle.	
DYNAMICS		
Coin Collection	Continued coin collection - either by colliding in the air or upon collision with bricks - results in the gain of extra 'lives' or '1-Ups'. Allowing for another attempt at play should players achieve a negative terminal state.	
NPC Kills	Chaining multiple collisions with NPCs - either through use of a kicked 'Koopa Troopa' shell or by jumping on their head - can result in increased score and extra lives.	
Chaining Jumps and Collisions	Given the deterministic nature of *Super Mario* games, chained collisions and well-timed jumps allow players to learn from mistakes quickly and discover more optimal solutions to levels.	
AESTHETICS		
Sensation	*The Super Mario* series is built to be visually appealing: adopting bright and colourful art assets. The aforementioned dynamics of the series are often reinforced through visual representation and charming audio effects.	
Fantasy	Given the mechanics exhibit exaggerated human traits, *Super Mario* games provide a make-believe scenario for players to explore.	
Challenge	*Super Mario* levels act as obstacle courses, with emphasis on avoiding pitfalls and enemies to reach the end within the time limit.	
Discovery	Many levels carry hidden areas for players to discover. This ranges from items hidden in question blocks to warp pipes that provide short-cuts. These short-cuts can range from portions of the same level to skipping entire worlds of play.	

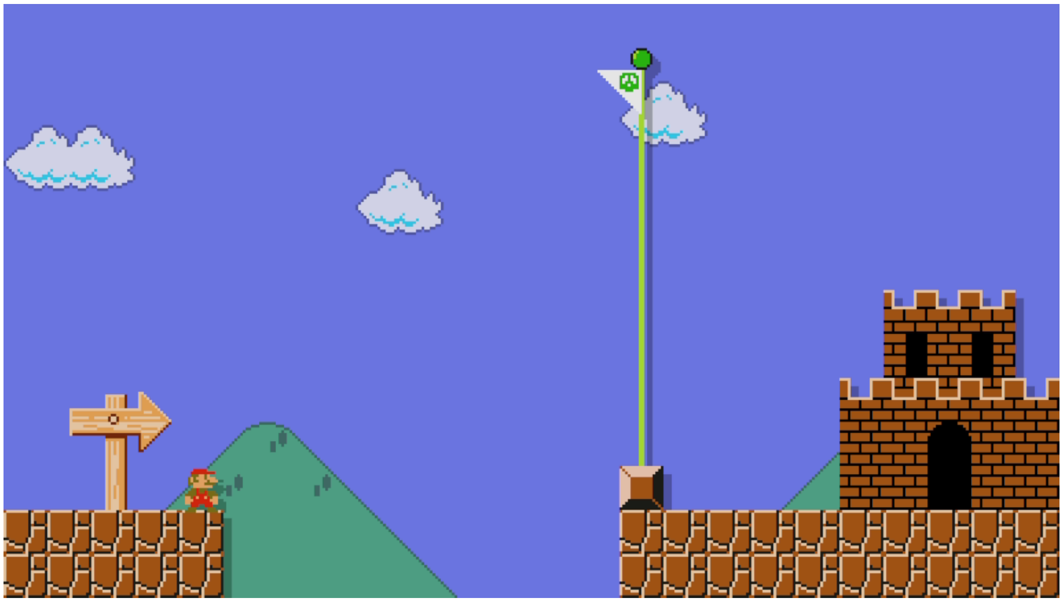


Figure 2: A representation of the design space afforded to human designers and procedural generators. With the notable gap between start point and flag the space within which gameplay experiences can be crafted.

The Design Space

We assert that in order for a level to be functionally complete, it must provide space within which the mechanics, dynamics and aesthetics found in Table 1 can be expressed. As such, we identify a simple reduction of a *Super Mario* level as follows:

Starting Point: Each level must provide a safe location for the Mario avatar to appear and allows players to assume control of the character.

The Challenge Space: A finite area of two-dimensional space after the starting point within which designers are able to place platforms, items, obstacles and NPCs.

End-Level Obstacle: The final 'chunk' of the gameplay space is the end-level obstacle that the player must cross in order to reach the 'level won' state.

In this representation, it is possible for players to achieve either terminal state through use of the provided mechanics. The 'challenge space' is where designers have sole input into the creation of *Super Mario* experiences: exploiting the dynamics and art style of the game to create a range of aesthetic qualities. This is represented courtesy of Figure 2 in which the challenge space is currently empty. Perhaps unsurprisingly, this image is taken from the *Super Mario Maker* level editor and is the 'smallest' *Super Mario* level that can be constructed whilst being functionally complete.

---

character with individual character design and dynamics.

[3] The full aesthetic taxonomy is as follows: Sensation, Fantasy, Narrative, Challenge, Fellowship, Discovery, Expression and Submission

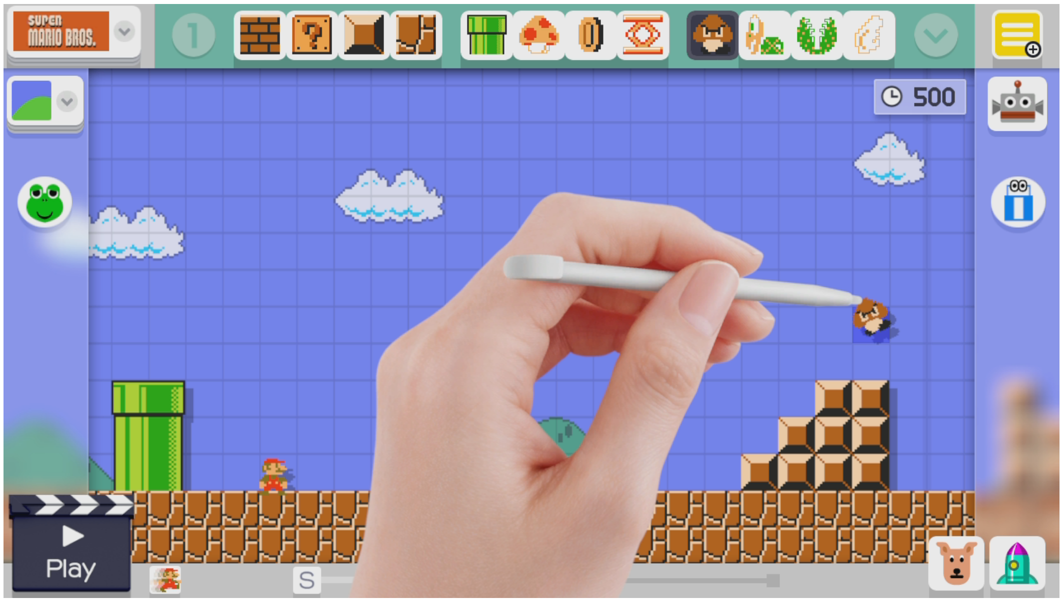


Figure 3: Screenshot of the editor in *Super Mario Maker* as a player attempts to place a 'Goomba' enemy onto the level.

Super Mario Maker

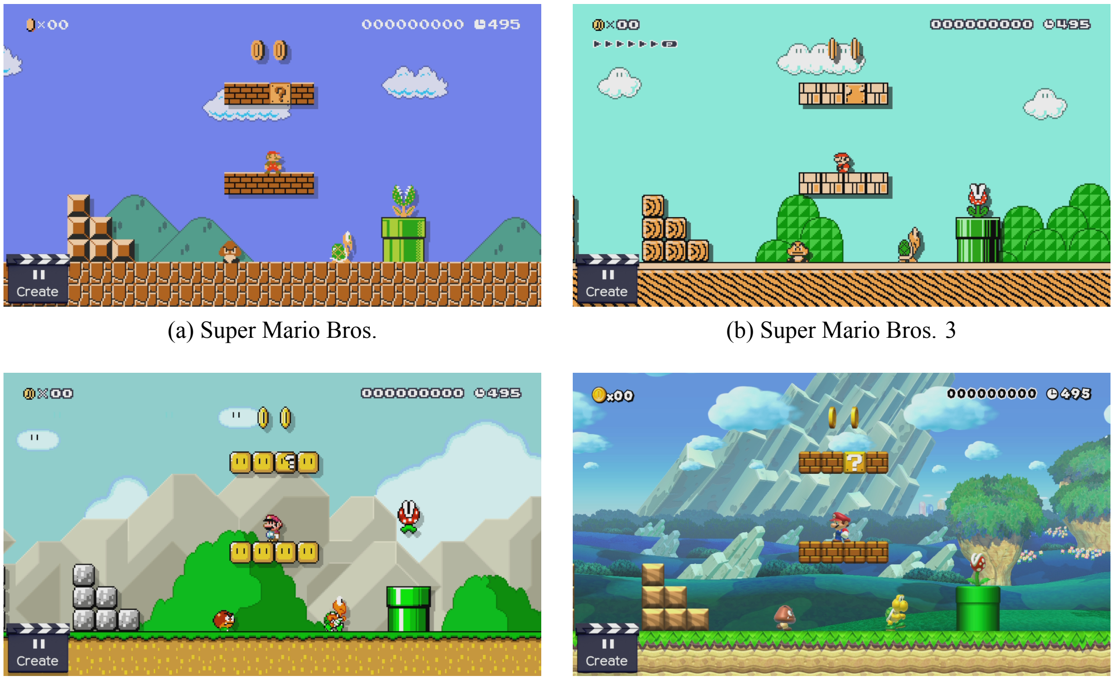
Super Mario Maker is a game creation platform on the Nintendo Wii U system inspired by the *Super Mario Bros.* series. The game utilises the touch-screen interface of the Wii U controller, allowing players to place terrain, bricks, items and enemies within the finite two-dimensional space of a level. The editor is highly responsive and flexible: allowing players to manipulate any and all objects in the game world and play-test levels at any time.

Players can create levels that adopt the art style of four games in the series: *Super Mario Bros.* (Fig 4a), *Super Mario Bros. 3* (Nintendo R&D4, 1990) (Fig 4b), *Super Mario World* (Nintendo EAD, 1991) (Fig 4c) and *New Super Mario Bros. U* (Nintendo EAD, 2012) (Fig 4d). While the series has largely retained the same core gameplay functionality, each game 'skin' will result in minor changes to the physics, gameplay mechanics and enemy behaviour to reflect the chosen title. While items in the editor may originate from a specific game in the series, players can place these items - such as brick types and enemies - in an anachronistic fashion: allowing for level configurations that would have been impossible in previous entries of the *Super Mario* series[4]. One further addition to traditional *Super Mario* design is the use of custom mushrooms that allow for the Mario avatar to be replaced by other Nintendo characters through use of *Amiibo* figures[5]. Once a designer is satisfied with their level, they must complete it by reaching the flagpole at the far end. Once achieved, the level can be published to an online service for fellow players to discover.

While the emphasis is on the creation of *Mario*-esque levels, *Super Mario Maker* provides

[4]For example, enemies such as the Bob-omb and Boo, which first appeared in *Super Mario Bros. 2* and *Super Mario Bros. 3* respectively, can be adopted in levels using the *Super Mario Bros.* aesthetic.

[5]A 'toys-to-life' platform that allows for use of specific assets in games through use of near-field communication (NFC).



(a) Super Mario Bros.

(b) Super Mario Bros. 3

(c) Super Mario World

(d) New Super Mario Bros. U

Figure 4: Screenshots of a 'traditional' design for a *Super Mario Bros.* level in *Super Mario Maker* having switched the 'skin' to one of four available in the editor.

functionality that is not reflective of *Super Mario* tropes. The level editor allows users to create a variety of novel interactions by relaxing the constraints of objects and their interactions with the game world. For example, enemies can be stacked atop one another, launched from nearby cannons or spawned from warp pipes. Similarly, constraints on power-ups are relaxed: with items such as the super mushroom - which provides Mario with a health boost and an increase in size - can be applied to almost any non-player character. These mutators are stackable, allowing for a variety of unique combinations for players to explore.

Procedural Generation Research In Super Mario Bros.

We take a moment to address the broad range of work that has been conducted within *Super Mario Bros.* and similar 2D platforming games. Please note this is by no means a complete survey of research in this area and would advise readers interested in this problem to consider the literature referenced throughout this section as merely a starting point for further reading.

Procedural Generation for Level Design

The body of PCG research focussed on *Super Mario* games was established largely by the first level generation track of the *Mario AI Competition*. The competition was reliant upon an open-source reproduction of *Super Mario*, entitled *Infinite Mario Bros.*, which adopted the mechanics and art assets of *Super Mario World*. *Infinite Mario Bros.* was initially extended for use as a benchmark for artificial and computationally intelligent algorithms [18, 17], but was subsequently extended once more to provide a consistent software API for level creation. Thus permitting the competition to expand its focus towards procedural level generation [19]. Submissions to the level generation track were required to adopt data from a sample level that players completed first and use it to inform the level generation process. The variety of controllers, detailed in [11] are driven by a variety of rule-based and evolutionary algorithms that construct levels through the structure and placement of platforms, items, blocks, obstacles and enemies.

One area that has proven highly active is an emphasis on the grammar-driven methods for level generation. Works detailed in [9, 10]. The focus of these works are reliant upon the adoption of design grammars that encapsulate the core functional aspect of Mario levels (platforms, cannons, hills, pipes, bricks, enemies etc.) and adopt search-based procedural generation techniques to find adequate level representations. These learning approaches place an emphasis on optimising against perceived engagement and/or frustration of players through player modelling approaches.

Outside of the *Super Mario* benchmark, there are similar works focussed on rule- or grammar-driven generation to be found in PCG literature. One notable body of work is the *Launchpad* game detailed in [14]. Launchpad is a game similar in design to *Super Mario Bros.* with many of the core mechanics albeit simpler in execution. The project aimed at identifying an alternative grammar for interpreting interesting gameplay sequences through use of rhythm and timing which would dictate the placement of platforms and other items in the design space. This work was subsequently followed by *Tanagra* which adopted many of the ideas from Launchpad to create a mixed-initiative level editor for a 2D platforming game: allowing designers to either interact with the AI designer or simply let it generate levels autonomously [13].

More recently, there has been an effort to explore how to generate levels by reverse engineering the 'Mario method': the implicit rules and constraints adopted in the creation of levels. This has been adopted in various forms, from the adoption of 'design patterns' to formalise axioms of level construction that can be used by level generators [4], to training Markov Chain-driven generators from an existing corpus of levels [15] to the use of YouTube playthrough videos to establish common level design traits [5].

Level Analysis and Classification

Outside of research focussed on building level generators, there is a growing body of work focussed on the analysis and classification of level design. Research in level analysis has placed focus on establishing core tenets of game design exhibited within the *Super Mario* series. This has largely been driven by research in 'design patterns': a formalism that allows for aspects of a games design at varying levels of abstraction to be conceptualised. Research detailed in [3] aimed to establish the core design patterns of the original *Super Mario Bros.* by establishing a set of design patterns that assessed the repeated adoption of gameplay conventions throughout the game. This resulted in a collection of different patterns that exist at varying levels of abstraction, from micro patterns that could be exhibited within one thin 'vertical slice' of the gameplay space - such as a solitary enemy - to 'macro' patterns that were reflective of significantly larger subsections of entire levels. While this work has been adopted as means for procedural generation systems to be adopted as noted previously in [4], it has also resulted in further discussion of the *Super Mario* design space. This is exhibited by work detailed in [16] which shows preliminary analysis of how specific patterns continue to be adopted in subsequent *Super Mario* games, while others loss favour or are evolved into more complex permutations.

Another element heavily related procedural content generation for 2D platformers is means to classify and even measure the quality of the levels. Work by Smith and Whitehead detailed in [12] has proven to be of immense value in more recent years of procedural generation research having established a range of potential metrics that can be used to assess a levels design. These metrics are focussed purely on the more functional aspects of these gameplay spaces with the aim to help measure the "expressive range" of generative systems. This has proven useful in comparing generative output on a functional level. Work detailed in [6] takes this approach by comparing a variety of levels from contemporary literature in *Super Mario* generation or similar problem domains. Such research in establishing metrics has been succeeded more recently by Smith and Canossa in [1] which introduces a larger set of metrics for measuring generated content. While the original publication was focussed more on the quality of generated content from a functional perspective, this work introduces a number of metrics aimed at establishing the aesthetic qualities of the crafted levels.

Non-Formalist Level Design in *Super Mario Maker*

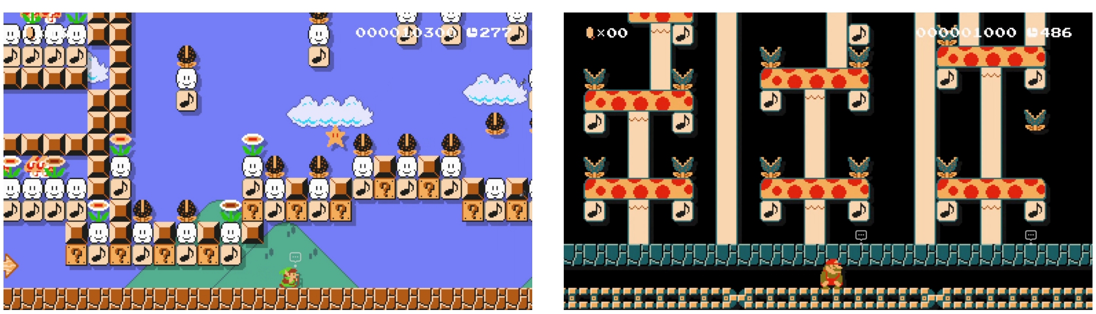
We believe that a significant number of levels designed within *Super Mario Maker* are a prime example of the non-formalist approach to game design discussed in [2]. We now provide a broad and far-from-complete summary of the types of content being created by the *Super Mario Maker* community. The content discussed was discovered through a brute-force incremental analysis of the most popular content generated for *Super Mario Maker* on a weekly basis between September and December 2015, resulting in assessment of 160 levels. This analysis focussed on the level design principles, adoption of existing dynamics and the aesthetic qualities of the content. From this we establish a number of common themes that significantly deviate from the *Super Mario* formula.

Performance Arguably the most popular form of level found is the automated or 'performance' level, typified by that shown in Figure 5. These levels remove the challenge of traversing the design space to the end-obstacle through an automated process that utilises the dynamics of the game and collisions with items such as springs, falling platforms and enemies. In many instances, should a player provide input to the Mario avatar, it disrupts the deterministic system the designer has crafted and results in a negative terminal state. These levels are perhaps best identified as 'Rube Goldberg' devices [20]: deliberately over-engineered to perform otherwise simple tasks[6]. These levels often adopt the expression aesthetic from the MDA framework as designers aim to create increasingly complex and vibrant performances which are awe-inspiring to witness.

Musical While similar to performance levels, these designs adopt 'music blocks'- that play a particular note when an object collides with them - in order to play a specific theme or song. A players input to these levels can vary, with some automating the movement

[6]The term itself is named after the cartoonist Rube Goldberg who popularised the notion.

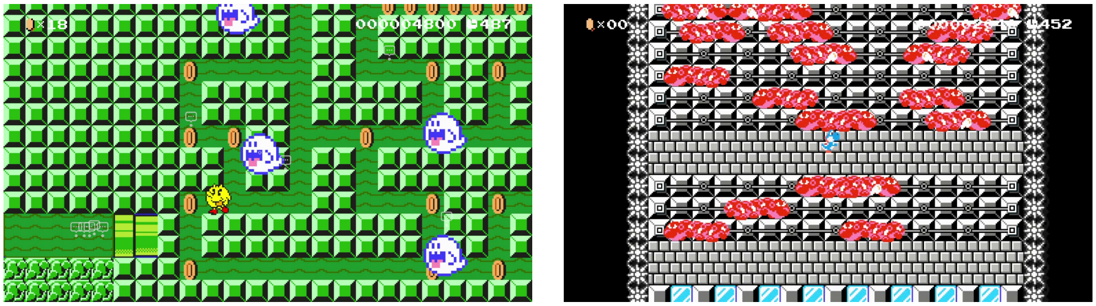
Figure 5: A performance level in *Super Mario Maker* in which designers crafted excessively complicated devices that navigate Mario from starting point to end-level obstacle without player input. [Level ID: 9525-0000-003B-F371]



(a) User-Driven

(b) Fully-Automated

Figure 6: Figure 6a is a musical that requires player-interaction, while Figure 6b is fully automated [Level ID's: 1385-0000-0072-E676 & 151F-0000-001B-A957].



(a) ‘*Pac-Man*’ (Namco, 1980) (b) ‘*Crossy Road*’ (Hipster Whale, 2014)

Figure 7: A particular trend in *Mario Maker* is to design levels that mimic the gameplay structure and mechanics of existing games. [Level ID’s: C70D-0000-002E-6128 & 04A3-0000-00D0-6571].

courtesy of a conveyor belt or through performance as denoted previously. Two common examples shown in Figure 6 are user-driven and fully-automated levels. Furthermore, while this is once again evidence of an expression aesthetic adopted, there is clear intent to illicit emotional responses from players. Almost all levels found aimed to reproduce themes from video games and other forms of popular culture, such as *The Legend of Zelda* (Nintendo R&D4, 1986) in Figure 6a and *Super Mario Land* (Nintendo R&D1, 1989) in Figure 6b.

Homage Homage levels aim to mimic the gameplay and level designs of existing video games. Such levels tend to be highly creative given the limitations imposed by the mechanics and dynamics of *Super Mario* games; requiring some ingenuity given the inspiration for levels such as those in Figures 7a and 7b are not 2D platformers.

Story These are levels aim to tell stories through intelligent use of art assets and heavily conform to the narrative aesthetic identified in [7]. Examples such as Figure 8 exploit well-known design tropes from existing games to reinforce the themes of the narrative.

Challenge Challenge levels are aimed at taxing the skill or intellect of the player. The former is reliant upon players intimate understanding of the level design, mechanics and dynamics of *Super Mario*. Meanwhile, skill-based challenges often vastly exceed the difficulty found in *Super Mario Bros.* games. While this adopts the previously established aesthetics of challenge and discovery, it clearly aims to illicit an exaggerated emotional response from players given the frustration and elation from continued experimentation until success. Meanwhile intellect-based challenges, such as that shown in Figure 9 present logic puzzles through use of existing art assets. Players must then manipulate the environment in order to ‘answer’ the proposed question: failing to answer correctly results in death.

Discussion

The subset of the popular content in *Super Mario Maker* discussed in previous sections highlights a broad range of levels that address a variety of aesthetics; with an aim to illicit a

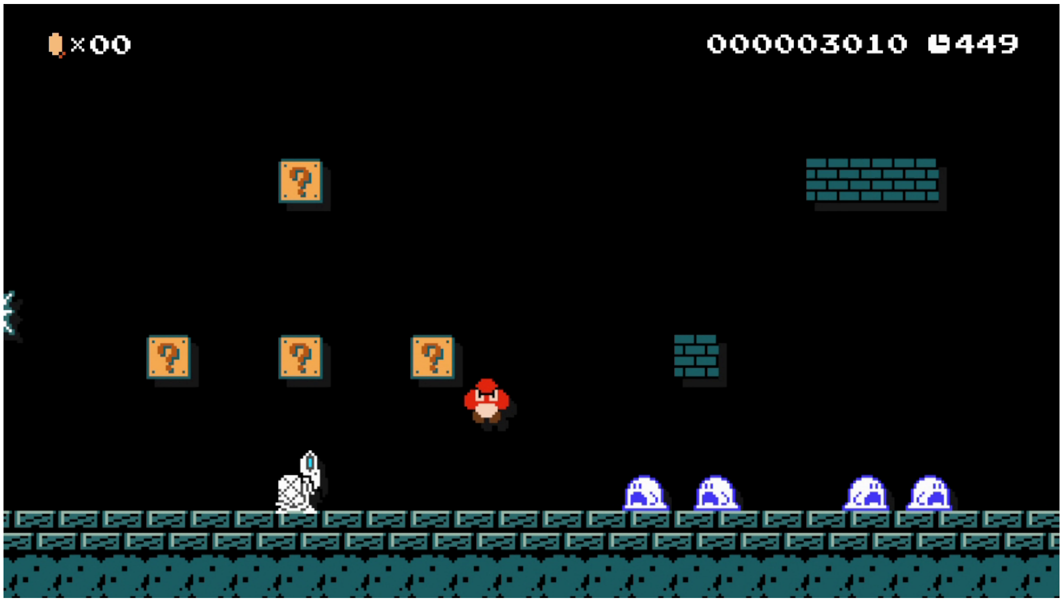


Figure 8: *The Goomba: A Sad Tale of Madness* is a level that deals with the futility of life from the perspective of a goomba NPC. Players must play *Super Mario Bros.* level 1-1 in reverse avoiding fellow goomba's who are now ghosts. [Level ID: BDAB-0000-0033-2697]

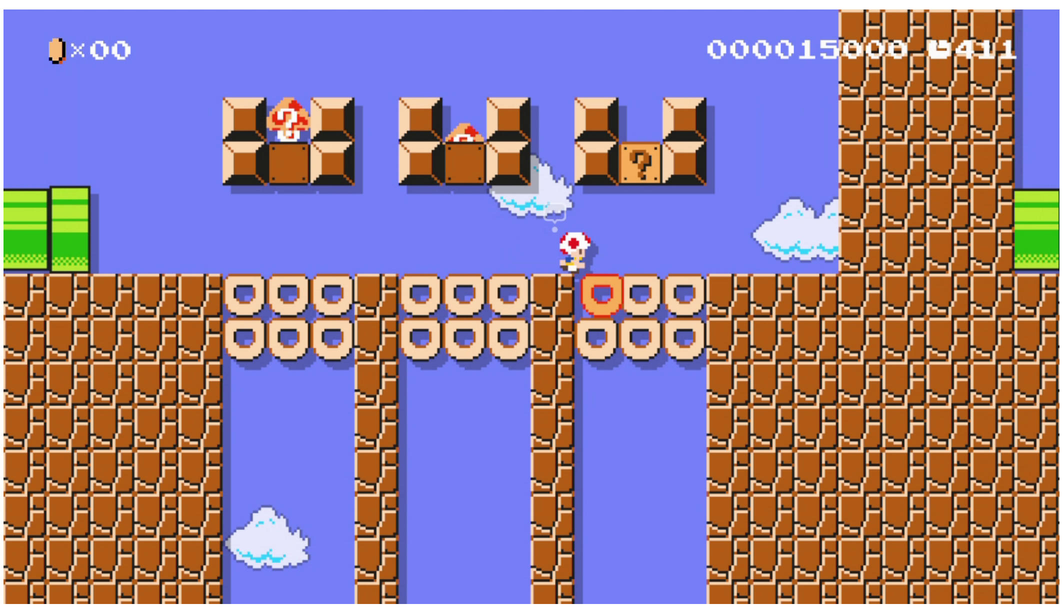


Figure 9: *Quiz: Who's In The Wrong Series?* is a level that exploits the Amiibo sprite collection to challenge a players knowledge of Nintendo characters and franchises. [Level ID: 7A60-0000-008F-A3DF]

variety of emotional responses from players. By creating, playing and up-voting these levels, the player-community is embracing "non-formalist" game design. While *Super Mario Maker* ensures the validity and playability of levels through use of the series mechanics and dynamics, it has led to a significant focus on aesthetics and the emotional qualities that designers can bring to this space. Designers find a broader range of aesthetics to choose from while adhering to the explicit mechanics and dynamics mentioned previously. This ultimately produces a framework for non-formalist game design within the confines of an established formalist system.

This content not only openly challenges the question of *'What is a Super Mario level?'* but the level editor itself permits its creation. Rather than establishing a more conservative view of what players *should* do within the design space, it openly asks what players *could* do within it. This is what makes this highly relevant to the procedural generation community interested in level design, given that in many respects it has adopted a more conservative view of the design space within *Super Mario*.

This is not to say that academic research in this field and more specifically the *Mario AI Competition* by any means enforced this view explicitly. Rather, it has been an unspoken and illicit agreement from within the academic community to focus on the functional aspects of *Super Mario* level design. Virtually all level generators discussed in our literature review adopt a functional pretence to the creative process. This places an emphasis on establishing rules for the placement of blocks, pipes and enemies to mimic that seen in *Super Mario* titles. It is evident from this particular vein of research, which continues to prove highly popular in the artificial intelligence community, that establishing rules of block placement is a largely solved problem. The emphasis now, as highlighted by works such as that found in [4] and [5] is on crafting level generators that reverse-engineer the 'Mario method'.

One immediate action is to broaden our focus on the assessment of generated levels to adopt more aesthetic metrics. The work detailed in [1] already makes steps towards this by establishing a set of suitable metrics - with future work for the author to quantitatively evaluate the sample set used in this paper against these metrics. At minimum, adopting functional and non-functional metrics when assessing the output of level generators should be considered where applicable. However, this may require that designers go back to first principles and question whether their level generators even factor these possibilities? The re-introduction of competition benchmarks akin to the *Mario AI Competition* would be of great value to the community should they place a greater emphasis on the aesthetic qualities of levels that can be crafted. Given the significant body of existing research in 2D platformer generation, expanding horizons within this space strikes us as one of the earliest opportunities to approach this non-formalist view of game design. Achieving generated levels akin to those described in the previous section veers heavily into the field of computational creativity, but would provide interesting benchmarks and metrics to strive towards outside of those previously established. The *Mario AI Competition* was evaluated based on whether levels proved 'interesting' to play. The community would benefit from removing such limitations and embracing a broader range of analysis (e.g. most exciting/difficult level, best story/performance etc.). Naturally this leads into complications of how to quantify levels that when dealing with more subjective metrics of evaluation.

Furthermore, there are opportunities present in providing an open and accessible tool-set for level generation that is more akin to the flexibility permitted in *Super Mario Maker*. The *Infinite Mario Bros.* benchmark, while suited for its original intent, lacks many of the functional and artistic assets required to replicate the kinds of levels discussed in this paper. While this benchmark has proven invaluable to artificial and computational intelligence research in recent years, it is arguably not suitable for addressing these new ideas.

CONCLUSION

In this paper we have argued that the proposal of aesthetic-driven level generation to be considered within the *Super Mario Bros.* problem domain. This would place a stronger emphasis on non-traditional forms of play and actively questioning the goals and motivations for players even in traditional and conservative gaming frameworks. This is in recognition not only of the significant body of work in PCG conducted in *Super Mario Bros.* but how it contrasts with the more aesthetic-driven works of human designers in *Super Mario Maker*.

Our aim is to establish not only that this form of work is acceptable, but should be embraced as a possible route for research in level generation. It is perhaps fitting that the first opportunity given to players to create levels within the framework of *Super Mario* rules creates a range of new aesthetic-driven content that is remarkably novel for the genre. From the wealth of research literature it is evident that, as a community, AI researchers know how to procedurally generate 2D platforming levels. As such, it is appropriate as researchers and level designers in our own right to challenge the implicit biases we ourselves have established and ask: *"What is a Super Mario level anyway?"*

ACKNOWLEDGMENTS

The author would like to thank Alex Zook for his contributions and remarks towards this paper and to our reviewers for their feedback during the peer-review process.

References

[1] Alessandro Canossa and Gillian Smith. "Towards a Procedural Evaluation Technique: Metrics for Level Design". In: *Proceedings of the 2015 Conference on the Foundations of Digital Games (FDG 2015)*. 2015.

[2] Michael Cook and Gillian Smith. "Formalizing Non–Formalism: Breaking the Rules of Automated Game Design". In: *Proceedings of the 2015 Conference on the Foundations of Digital Games(FDG 2015).*

[3] Steve Dahlskog and Julian Togelius. "Patterns and procedural content generation: revisiting Mario in world 1 level 1". In: *Proceedings of the First Workshop on Design Patterns in Games*. ACM. 2012, p. 1.

[4] Steve Dahlskog and Julian Togelius. "Patterns as objectives for level generation". In: *Proceedings of the Second Workshop on Design Patterns in Games, the 2013 Conference on the Foundations of Digital Games (FDG 2013).* ACM, 2013.

[5] Matthew Guzdial and Mark O Riedl. "Toward game level generation from gameplay videos". In: *Proceedings of the FDG workshop on Procedural Content Generation in Games*. 2015.

[6] Britton Horn et al. “A comparative evaluation of procedural level generators in the mario ai framework”. In: *Proceedings of Foundations of Digital Games* (2014).

[7] Robin Hunicke, Marc LeBlanc, and Robert Zubek. “MDA: A formal approach to game design and game research”. In: *Proceedings of the AAAI Workshop on Challenges in Game AI*. Vol. 4. 2004.

[8] Sergey Karakovskiy and Julian Togelius. “The mario ai benchmark and competitions”. In: *Computational Intelligence and AI in Games, IEEE Transactions on* 4.1 (2012), pp. 55–67.

[9] Noor Shaker et al. “Evolving levels for super mario bros using grammatical evolution”. In: *Computational Intelligence and Games (CIG), 2012 IEEE Conference on*. IEEE. 2012, pp. 304–311.

[10] Noor Shaker et al. “Evolving Personalized Content for Super Mario Bros Using Grammatical Evolution.” In: *AIIDE*. 2012.

[11] Noor Shaker et al. “The 2010 Mario AI championship: Level generation track”. In: *Computational Intelligence and AI in Games, IEEE Transactions on* 3.4 (2011), pp. 332–347.

[12] Gillian Smith and Jim Whitehead. “Analyzing the expressive range of a level generator”. In: *Proceedings of the 2010 Workshop on Procedural Content Generation in Games*. ACM. 2010, p. 4.

[13] Gillian Smith, Jim Whitehead, and Michael Mateas. “Tanagra: A mixed-initiative level design tool”. In: *Proceedings of the Fifth International Conference on the Foundations of Digital Games*. ACM. 2010, pp. 209–216.

[14] Gillian Smith et al. “Rhythm-based level generation for 2D platformers”. In: *Proceedings of the 4th International Conference on Foundations of Digital Games*. ACM. 2009, pp. 175–182.

[15] Adam James Summerville, Shweta Philip, and Michael Mateas. “MCMCTS PCG 4 SMB: Monte Carlo Tree Search to Guide Platformer Level Generation”. In: *Eleventh Artificial Intelligence and Interactive Digital Entertainment Conference*. 2015.

[16] Tommy Thompson. “The Fine Line Between Rehash and Sequel: Design Patterns of the Super Mario Series”. In: *Proceedings of the Fourth Workshop on Design Patterns in Games, the 2015 Conference on the Foundations of Digital Games (FDG 2015).* 2015.

[17] Julian Togelius, Sergey Karakovskiy, and Robin Baumgarten. “The 2009 mario ai competition”. In: *Evolutionary Computation (CEC), 2010 IEEE Congress on*. IEEE. 2010, pp. 1–8.

[18] Julian Togelius et al. “Super mario evolution”. In: *Computational Intelligence and Games, 2009. CIG 2009. IEEE Symposium on*. IEEE. 2009, pp. 156–161.

[19] Julian Togelius et al. “The mario ai championship 2009-2012”. In: *AI Magazine* 34.3 (2013), pp. 89–92.

[20] Maynard Frank Wolfe and Rube Goldberg. *Rube Goldberg: Inventions!* Simon and Schuster, 2000.