Analyzing Emergence in Pac-Man's Mazes with Systemic Reverberation

Jared Pettitt, Nathan Altice
University of California, Santa Cruz
1156 High St.
Santa Cruz, CA
jpettitt@ucsc.edu, naltice@ucsc.edu

ABSTRACT

To better formally analyze and design emergent gameplay, the authors define *systemic reverberation*. When players' interactions generate unique game states and they understand the relationship between their interactions and those consequent states, they are sensing systemic reverberation. Systemic reverberation describes how players are able to "echo-locate" through the complex possibility spaces generated by the rules of a game, as their interactions *reflect* and *absorb* across the topology of the possibility space. Further, by conveying unique states in response to particularities of player interactions, systemic reverberation conveys to the player that their choices "matter." Conversely, when interactions or sequences of interactions have less of an understandably unique effect on game state, players' interactions and their details are being *absorbed* by the game system rather than reflected. To illustrate systemic reverberation, we synthesize and apply existing scholarship on game systems, as well as carry out a comparative formal analysis using *Pac-Man* and *Pac-Man Championship Edition*. In so doing, we develop a list of clear, practical strategies to encourage systemic reverberation (and consequently, emergent gameplay) in game design.

Keywords

game systems, emergent gameplay, systemic games, formal analysis

INTRODUCTION

Systemic reverberation describes how a game's continual state changes and consequent feedback convey the relationship between a player's interactions and the game's possibility space, i.e. the complex, many-dimensional, abstract space produced by a game's rules, which represents every possible permutation of game state. Like a game's difficulty, systemic reverberation is a subjective quality occurring as a result of the combination between a game's formal design and its feedback mechanisms (audio, visual, tactile, etc.). The term is defined and applied here as a tool to understand how a game's design might afford or constrain emergent gameplay (expressive player interaction and/or scenarios). Put simply, systemic reverberation describes how a game conveys the contour of the terrain, its boundaries, and paths through its possibility space using player interaction (as input) and feedback (as output). As an interaction reverberates through the possibility space, details of that interaction may be reflected or absorbed across parts of the game system. Absorption occurs when a game's feedback mechanisms fail to convey relationships between interactions and game state, or when possible sets of interactions leading to a given game state overlap and conflate, such that the player

Proceedings of DiGRA 2025

is less able to discern the possibility space they're navigating and/or how their choices relate to their trajectory through that space.

FORMAL TERMS AND THEIR CONTEXTS

Game Systems and Systemic Games

Like many terms used in game design, a "game system" is defined across industry, academia, and popular discourse in multiple overlapping (and sometimes contradictory) ways. Christo Nobbs, a Senior Technical Game Designer specializing in systems, says that game systems "should have a clear function, hold data, and be modular… [pieces] of functionality that can run recursively through the game" (2021). In *Game Mechanics: Advanced Game Design*, Ernest Adams and Joris Dormans describe systems as "usually [having] many interacting parts. The behavior of individual parts might be easy to understand; their rules might be simple. However, the behavior of all the parts combined can be quite surprising and difficult to foresee" (2012). Benjamin Marshalkowski of the *Board to Bits Games* YouTube channel attempts to address what he perceives as a lack of "adequate definition" of game systems, defining them as "a combination of multiple mechanics, or other, simpler systems, that take in inputs and result in an output" (2018).

While the above references are relatively recent, designers have been theorizing around systemic behavior in games since the early history of commercial digital games. In his pioneering work of game formalism, *The Art of Computer Game Design*, designer Chris Crawford described that the ambiguity of said concepts left game designers "no well-defined set of common terms with which to communicate." (1984) However, Crawford's definition of a system — "a collection of parts that interact with each other, often in complex ways" — was one of several related terms comprising a larger definition of games, what he described as, "a closed, formal system that subjectively represents a subset of reality" (1984). By "closed" and "formal," Crawford meant that games were "complete and self-sufficient" and adhered to "explicit rules," but the salient point is that all games were systems of some kind. And systems, in turn, required multiple parts with defined relationships that might exhibit complexity.

By defining all games as systems, Crawford drew a distinction between games and other forms of playable/interactive media, namely, puzzles, stories, and toys (see Figure 1). Compared to the responsive, rule-oriented mechanical interactivity of games, Crawford argued, puzzles have manipulable mechanics and rules but do not respond to the player; stories proceed linearly without mechanics or manipulability; and toys are freely manipulable without rules or mechanics. Consequently, Crawford stressed the significance of interaction in games: "Interaction injects a social or interpersonal element into the event…The key distinction between a game and a puzzle is the difference between creating your own solution and discovering the designer's solution." (1984)

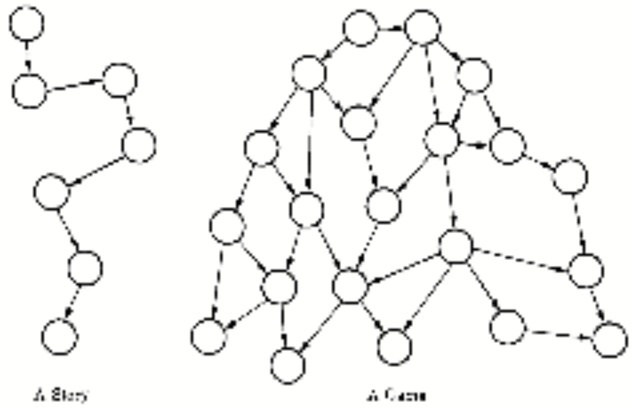

Figure 1: Crawford illustratively compares the interactive structure of a game to the non-interactive structure of a story (1984).

This same distinction, cast with different terms, is echoed widely in literature concerning systemic game design. Michael Sellers, in *Advanced Game Design: A Systems Approach* draws a similar distinction between games that are "content-driven" and those that are "systemic." Where content-driven games have players "[progressing] along a path laid out by the game designer," systemic games "use complex interactions between parts to create the game world… [setting] up the conditions that will guide the player in creating their own path–one of a large number in a vast game-space." When a game's rule structure is linear, progress doesn't occur through player expression, but through the player carrying out some predefined process(es) that the game is prompting. When a game's rule structure is less linear, when its parts are connected in a web rather than a chain, the game is better able to support the types of interpersonal interactions Crawford claimed to be central to games.

Sellers defines systems as "a set of parts that together form loops of interaction between them to create a persistent whole. The whole has its own properties and behaviors belonging to the group but not to any single part within it" (2017). Sellers describes games' rule structures as being built of parts and behaviors. Parts are often analogized in game design texts as a game's "nouns," whereas behaviors are the "verbs" that nouns do. In more concrete technical terms, parts are like a game's computational objects which encapsulate data, and behaviors are the methods used to interface with those objects. In *Pac-Man* (Namco 1980), for instance, the parts are Pac-Man, the ghosts, the dots, the power cookies (Iwatani 2017), fruit targets, and the enclosing level. *Pac-Man*'s behaviors comprise every interaction between parts, including parts' interactions on themselves (see Figure 4).

Parts and behaviors can be composed into looping relationships, and those looping relationships can carry transitive effects. When a part initiates a behavior that causes an effect to propagate through two or more parts which then feed back to the initiating part, those constituent parts and behaviors form a transitive loop. Presence of transitive loops is a key way to delineate between a system and what Sellers calls a "complicated process." The former contains one or more transitive loops; the latter is a sequence of specific states which are moved through in succession (2017). Where a Rube Goldberg machine carrying a ball across a series of mechanisms is a complicated process, a pinball table flinging a ball around in response to bumpers, flippers, and other components, is a system (see Figure 2).

Figure 2: Comparing Rube-Goldberg machines *(source: trebound.com, Wikimedia Commons)* and pinball tables *(source: SunofErat, Wikimedia Commons)* illustrates the difference between what Sellers calls "complicated processes," and systems.

Sellers refers to games displaying these qualities as systemic games. Rather than simply behave with complexity, systemic games' parts and their looping interactions combine into something greater than than their sum. In systemic games, the parts and behaviors defined in a game's rule structure create a metaphorical, nonlinear state space that the player traverses—another way of describing the phenomenon of the game responding to the player. This space results from the game's rule structure, emerging as the combination of all possible game states afforded by its parts and interactions. Mechanics furnish the player with the ability to traverse this dynamic space (whose bounds remain a static quality of the game's overall design). When a player interacts and the game responds, the player is navigating the space using mechanics. Sellers refers to this relationship between game and player as being "structurally coupled," arguing that, "if a game is systemically designed, it will have a sufficiently broad and diverse state-space that it can adapt to the player as the player adapts to it" (2017, 82).

Possibility Spaces

In game design literature, such state-spaces are also referred to as probability spaces or possibility spaces. Designer Will Wright, best known for systemic simulation games like *SimCity* (Maxis 1989)**,** describes possibility spaces as "many-dimensional spaces the player moves through, each point an individual permutation of possible game states." In his GDC 2003 talk, "Dynamics for Designers," Wright presents abstract depictions of *The Sims*' (Maxis 2000) and *The Sims Online*'s (Maxis 2002) possibility spaces (see Figure 3). Wright explains the paths that players generally move through in *The Sims*, tending to get "trapped in local maxima" according to social or material progress, needing to balance the two in order to succeed (2001). If players focus too much on the social relationships or the materialistic goals of their Sims, they proceed further towards one extreme of the possibility space. The topology of the space is such that it is difficult to escape these local maxima without the player pulling back their narrow focus. In other words, the player's current location within the possibility space does not provide them with the mechanisms necessary to escape; the slope is too steep. This multi-dimensional progress is mediated by the complex interrelationship of all of the elements in the game working in tandem with the player continually, dynamically moving the player across the possibility space (and not, as these simplified diagrams may suggest, across three simplified axes of numeric material, social, and success values).

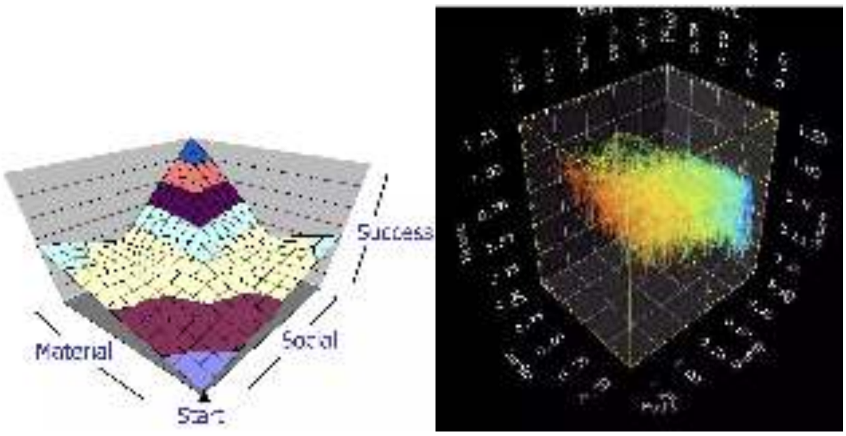

Figure 3: Abstract representations of the possibility spaces of *The Sims* (left) and *The Sims Online* (right). The left figure shows progress through The Sims according to different measures of success, where players can get "trapped in local maxima" (left and right back edges) of social and material success. The right figure shows many players' paths through *The Sims Online*, according to social, monetary, and house progress, forming a cloud that describes the game's possibility space (Wright 2001).

Game designers must intuit how rules and objects combine to form a space that the player traverses and design the mechanics they use for this traversal. The "game" takes place as the player's position in the possibility space changes, in response to the player-game interaction. In *Rules of Play*, Salen and Zimmerman describe the task of game design as one of "second-order design" (2003), which Zubek echoes: "[Game design] is a hard, second-order problem, as we are twice removed from the end goal. We are not simply crafting a static object; we are creating a dynamic machine which will behave in some way, and this behavior is what we hope players will enjoy" (2020, 7). Game designers must specify parts, attributes (data comprising a part's state), and behaviors that form an abstract architecture, which the player negotiates using to their mechanical affordances. Zubek relates the second-order design he describes to state spaces, saying that "...a game is a series of game state transitions over time." (2020, 46-47). Similarly, game system designer Marta Fijak argues that the game designer is "responsible for each state" the player moves through, as those states communicate meaning to the player (2021). Even if interconnected systems allow a player to enter states that the designer didn't anticipate, the fact that they are in that state, as well as the way that they got there, teach them about the game's behavior.

As a player proceeds through a game, they navigate the abstract topology of the game's possibility space and take in information about the game's behavior at every state. Meaning arises not only from the design of the games' parts and behaviors; it also arises from the space formed by their potential interactions. This space conveys meaning to the player, as does the way that the game responds to player interaction (how the topology of the space allows a player to traverse). Adams and Dormans refer to the player's path through a game's possibility space as their *trajectory*, describing both a game's possibility space and possible trajectories through that space as "emergent properties of the game rule system." (2012, 27). Viewing gameplay as a relationship between a game's possibility space and the player interactions it affords means half of the game's meaning derives from the possibility space formed by the game's parts and behaviors, and the other half of the game's ability to mean derives from how that space's topology shapes player behavior, their trajectory. **In a systemic game, a complex possibility space emerges from interactions between a game's parts, while trajectories emerge from this space's dynamic interactions with the player(s).**

Emergence

Games scholar Jesper Juul also stresses a similar distinction to Crawford's "puzzles and games," and Sellers' "content-driven and systemic [games]," in *The Open and the Closed* Juul categorizes games into those with "relatively simple rules but much variation," (emergence) and those with "elements can only interact in a limited number of predesigned ways," (progression) (2002). This echoes Crawford's puzzles/games dichotomy and situates emergence as a concept related to a game's possibility space and players' traversal through it. As players interact in particular ways and are carried to particular locations in the possibility space contingent on those behaviors, they experience emergent gameplay.

From the perspective of understanding games as possibility spaces, emergent gameplay is the phenomenon of the player understanding that specific interactions lead to specific outcomes within a "sufficiently broad and diverse [possibility space]," perceiving the game as adapting to their behaviors (2017, 82). A game's possibility space is an emergent property of its design, and the trajectories players take through it are emergent properties of the relationship between those players and that possibility space. Accordingly, systemic games are games that produce emergent states and allow players to perceive and interact with those states. However, games which contain systemic and emergent elements can also have effects, both in the short- and long-term that undercut or nullify these emergent elements.

To illustrate: Over the past 40 years, high-level *Pac-Man* play has reduced from a complex game of cat-and-mouse to a feat of rote memorization and endurance, where world-record players execute from a set of constrained patterns, rendering *Pac-Man*'s emergent ghost movement mechanical, predictable, and irrelevant (besides making players wait). At the moment, there is no way to describe such a gradual, narrowing reduction of trajectory space (and design strategies to mitigate it) without explaining it fully in context. Describing these complex effects requires considering a possibility space, its constituent parts and behaviors, and how those relate to the available trajectory space. We need language in order to describe the game design decisions and techniques that lead to systemic, emergent gameplay, which can also acknowledge that aspects of design can also lead to convergences in player behavior that decrease perceived emergence.

SYSTEMIC REVERBERATION (AND ABSORPTION)

Systemic reverberation describes how a player's mechanical behaviors reverberate (reflect and absorb) across the topology of a game's possibility space. Less metaphorically, it describes how a game reflects the details of player interaction in clarity, frequency, time, and/or distance, in its state changes and attendant feedback. Effects may be more or less specific to a particular sequence of interactions, may occur at different rates, may come sooner or later, and may be affecting a visually or semantically distant part. Systemic reverberation emanates from the topology of a possibility space, feeding back to the player, informing them how various parts of the game system respond to different actions. Variety and specificity in systemic reverberation demonstrate to the player that their actions "matter," and necessitate a "sufficiently complex" possibility space.

Systemic reverberation is what allows players to use their mechanical behaviors to echo-locate through the multi-dimensional topologies generated by games' rule structures. Interrelated parts in a game's system generate a non-linear possibility space through their looping behavior. Variety in those parts and their behaviors increases the heterogeneity of the space. Systemic reverberation conveys this complex space to the player when they interact using mechanics, allowing them to "feel" the possibility space they are navigating.

In pinball, the parts, behaviors, the possibility space they describe, and the trajectory a player takes through that space are physically literalized and observable. When a player pulls the launcher and sends the ball ricocheting across board, they can watch their trajectory play out and see how the specificities of their interactions can move the ball to different parts in the possibility space. When a player understands how hitting a bumper at different angles or speeds yields differing outcomes, the player is able to start formulating plans on how to use the bumpers based on their emergent understanding of the system. This understanding is provided to them by the systemic reverberation caused by the mechanics (the player's flipper hit) and feedback (how the pinball [player] perceivably moves across the system of the game, in this case visually, physically, literally).

Absorption describes the decay of a game's systemic reverberation, specifically how the architecture of a game's possibility space might cause less reverberation and/or reverberation to be less descriptive (specific). This happens when details of player interaction do not reflect further into other parts of the game system. When different mechanical interactions yield equivalent results, the game communicates that it does not behave with particular specificity toward those interactions, making them "matter less." This effect implies a homogenous response, communicating either a boundary of the possibility space or nothing (ambiguity). Absorption can be the result of a less complex space or a more homogeneous one; to the player, they "read" the same way. **While a game may contain many interconnected systems that form a complex possibility space, if that space does not reflect the specificities of players' mechanical interactions through perceivably unique trajectories (continual game state changes), then the space feels (and effectively is) smaller and more "homogenous".**

One particularly absorptive playfield feature in pinball is the "kick-out hole," (The Internet Pinball Machine Database, 2000) which catches the ball and then launches it in a predetermined direction with a predetermined velocity. While the ball accumulates momentum as a result of sequences of interactions with components, the player builds knowledge about how flipping the ball at different speeds and times can send the ball on different trajectories. When a ball lands in a kick-out hole, all prior interactions are absorbed. The ball's momentum and velocity, cumulative results of the sequence of player actions and components' reactions over time, are completely absorbed by the kick-out hole as it halts then ejects the ball along a predetermined trajectory. The player is no longer learning about how any of the actions they took before reaching that hole are playing out; only how the ball might react after having reached the kick-out hole. This homogenizes the possibility space of the pinball table, acting as a sort of "wall." Any set of interactions that at some point passes the ball over the kick-out hole now play out predictably, communicating less about how player behavior affects the system.

Systemic reverberation is complicated by a game's feedback. It's not only whether or not unique states are created, but also whether or not the player is able to interpret these changes and their relationship to their precipitating interactions. Drawing from Noah Wardrip-Fruin's operational logics (2020), systemic reverberation understands meaning in games as arising from the combination of abstract processes and communicative roles, communicating continuous state change (abstract processes) through feedback (communicative roles) in response to player interaction. Systemic reverberation asks how a game responds to a player's interaction with respect to the range of how a given game can respond (its possibility space) and what play the game affords (the trajectory space).

Even small changes in a game's design can dramatically change the volume and topology of its possibility space. Similarly, small changes in its feedback can dramatically change the way the game is understood. In a pinball table, for example, adding and removing components, or even minute changes in the placement of components, alters the dimension of the game's possibility space and how players navigate it. These relationships aren't always straightforward, because they need to be thought of in totality, in holistic reference to the rest of the game. A pinball game affording the player the ability to earn more balls through skilled play seems like a straightforward way to increase both the range of possible player experiences and the size of the possibility space, i.e., better players can play for longer, increasing the diversity of scores and possible combinations of state. However, if the game is not sufficiently complex such that the extra balls do not allow the player to do anything new (enter new permutations of game state), then the volume of the space is increased but it is not diversified by any new topology. The space is large, repetitive, and homogeneous; a player cannot differentiate between permutations of state that are functionally and representationally identical, and why should they?

COMPARATIVE ANALYSIS: *PAC-MAN* AND *PAC-MAN CHAMPIONSHIP EDITION*

To illustrate how changes in game design and representation can change a game's possibility space, we compare *Pac-Man* (Namco, 1984) and *Pac-Man Championship Edition* (Namco, 2020). *Pac-Man Championship Edition* (*CE*) is similar in design to the original, and the games' similarities allow us to analyze how their differences relate to changes in their expressions of systemic reverberation. Certain structural and mechanical changes in *CE* cause its possibility space to be larger and more complex, and its strict time limit combined with lack of level changes mean that mechanical interactions are absorbed continuously, at a much slower rate than in *Pac-Man*. In other words, the longer you play *Pac-Man*, the less it feels like what you do/what you've done matters; changes in *CE*'s design inverse this progressive narrowing of trajectory space.

Pac-Man

In *Pac-Man*, the player moves Pac-Man through a procession of levels by eating every dot and power cookie inside each level's maze. As they do this, ghosts alternate between chasing Pac-Man around the level and "scattering," each of the four ghosts moving to one of the corners of the map for a predetermined period of time. When Pac-Man eats a power cookie, the ghosts become scared, turning blue and moving away from Pac-Man; if a ghost is scared, Pac-Man can eat it for points, causing it to travel back to the center of the screen (as eyes without a ghost body), reappear as a ghost, and slowly exit back into the maze in a queue with other previously-eaten ghosts. When a certain amount of time passes after eating a power cookie, the ghosts cease being scared and return to chasing Pac-Man. If the player eats successive ghosts without running out the "scared timer", up to ghosts four total per cookie, each ghost is worth double the points of the last. "Fruit targets", which can be eaten for points, appear twice per level at dot-eating thresholds and disappear after a certain amount of time. Pac-Man's speed, the ghosts' speed, and the length of time ghosts remain scared are all attributes determined by the level. This means that speed changes occur at the start of levels two, five, and 21.

When the player makes a turn in *Pac-Man*, they are given a small window of time before and after reaching the center of an intersection to input direction, causing Pac-Man to exit the intersection at a slightly variable speed. There is no feedback that indicates this is happening in the game, making the effect difficult to perceive for casual players. This means that when Pac-Man reaches an intersection there are around twenty possible outcomes, depending on which direction Pac-Man is facing: going straight, various speeds of turning left or right, turning around, or running into a wall.

As the ghosts chase Pac-Man, the accumulation of his positions over time creates a game state that is uniquely reflective of player input choices. In *Pac-Man*, Pac-Man's movement reverberates into the game state, effecting Pac-Man's position, points, dots and power cookies remaining, which fruit targets have been eaten, how many lives are left, current level, length of time ghosts remain scared, as well as ghosts' speed and positions. When Pac-Man eats all dots and power cookies, the player moves to the next level, where the maze has been replenished with dots and power cookies, there is potentially a speed change, and the game begins anew, carrying forward only points and lives from previous levels.

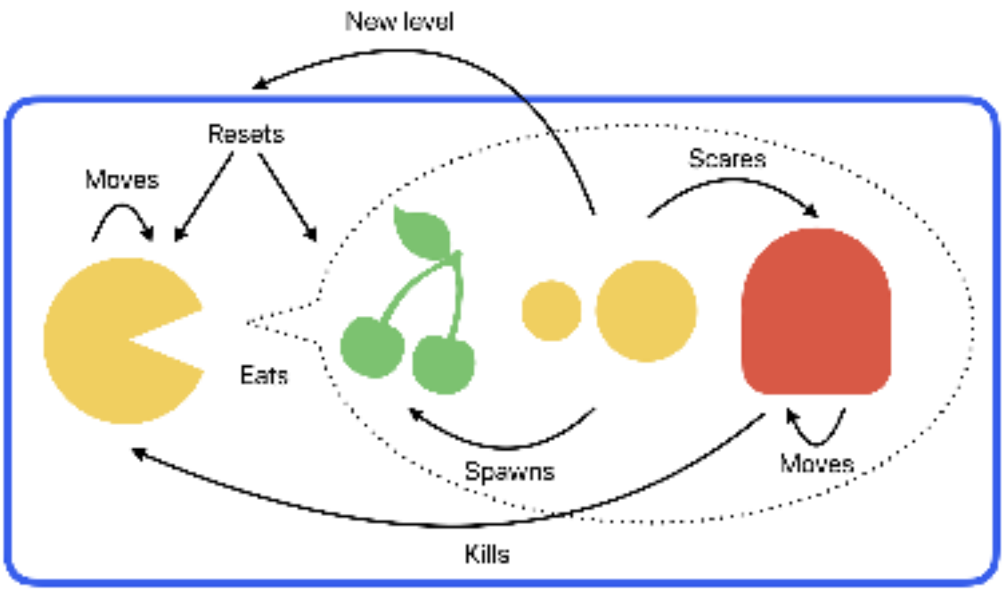

Figure 4: A simplified diagram of *Pac-Man*'s rule structure, showing its parts and how they relate to one another with behaviors.

Pac-Man Championship Edition

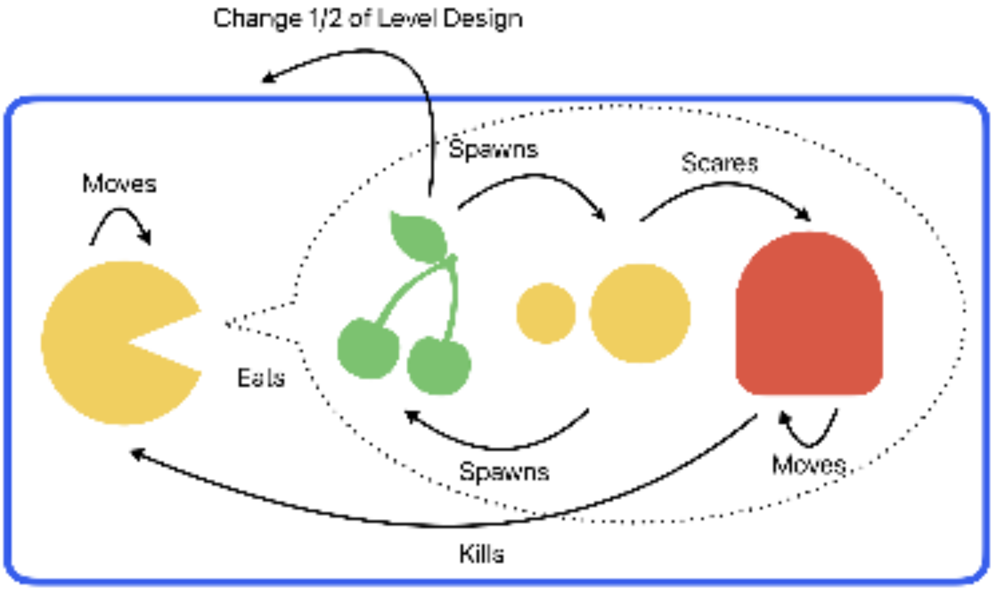

Figure 5: A simplified diagram of *Pac-Man Championship Edition's* rule structure, showing its parts and how they relate to one another with behaviors. Note the differences from Figure 4.

CE makes a number of small changes to *Pac-Man's* rules, relating more parts to fruit targets (see Figure 5). In *CE*, the player moves Pac-Man through a two-sided maze, eating two different patterns of dots and power cookies. Once a pattern is eaten, a fruit target appears on the opposite side of the screen and does not disappear until eaten. Once Pac-Man eats a fruit target, the design of the maze on the opposite side reconfigures into a new level design, and a new pattern of dots and power cookies appears (see Figure 6). These level designs are iterated as Pac-Man eats fruit targets, wrapping back to the initial design after ten permutations. Each side of the maze iterates separately. Each time Pac-Man eats a fruit target, the speeds of Pac-Man, the ghosts, and the length of the scared timer increase (at differing rates).

Figure 6: Different permutations of maze configuration, dots, and power cookies in Pac-Man Championship Edition which change in response to Pac-Man eating fruit targets.

In *CE*, speed-ups occur whenever Pac-Man eats a fruit target. This means that speed values change much more frequently in *CE* than in *Pac-Man,* and within a smaller window of time. A player that gets a perfect score in *Pac-Man* will experience three speed changes in around four to six hours. A top-scoring player in *CE* will experience around forty speed changes, continuously over five minutes. What this means is that a player in *CE* feels a constant, gradual increase in speed, and understands that their own movement is causing these changes. This relationship expands *CE*'s possibility space and the more gradual, subtle change means that the player must interpret and react to these changes in a tighter loop.

This granularity in speed further affects *CE*'s cornering mechanic, where speed increases mean less reaction time for quick turns. This causes the players' turn speed to naturally fluctuate, adding more player-determined chaos to the system. Similarly, *CE's* version of *Pac-Man*'s cornering both increases the number of potential outcomes and signals what is happening more clearly through visual feedback (see Figure 7). This added granularity and clarity not only diversifies the game's possibility space, allowing for movement to reverberate further and carry more meaning, but it also allows the player to understand cornering as one of the tools they use to navigate the game's possibility space.

Figure 7: Cornering has visual feedback in *Pac-Man Championship Edition*: Sparks erupt from Pac-Man's friction against the wall as he leans against it, prepared to turn right.

Most importantly, *CE* has no level changes. Where *Pac-Man's* level change stops the player, all the ghosts, and repositions them into a preconfigured setting to start again, *CE* has no such reset. This, combined with the evolving level design, means that rather than setting through a sequence of repeated mazes for an indeterminate amount of time, the player is instead navigating through a large set of possible level configurations, all dictated by their own movement. This both diversifies the possibility space such that emergent gameplay (and systemic reverberation) is made more possible, as well as enhances the player's sense that their movement choices matter.

"The Perfect Game"

In *Pac-Man*, the maximum score possible is 3,333,360 points. The only way to attain this score is to not only play through each of the game's 255 levels, but also to get every fruit target, never lose a life, and eat all four ghosts for every power cookie. Levels 21 to 255 play identically. Upon reaching level 256, the game can no longer display the map correctly, and the player must use the remaining lives to get sets of now-invisible dots (due to display errors) that re-appear after each death. A perfect game of *Pac-Man* becomes an endurance test that can last up to five hours. Repetitive paths are devised and mastered to ensure Pac-Man gets every fruit target, takes advantage of every scatter phase, and maximizes scoring combos. At higher levels, maximizing combos requires players to watch and wait for long periods of time for the ghosts to "line up" correctly so Pac-Man can eat all of them in the one or two seconds allotted (Pittman 2015). To a top player of *Pac-Man*, any emergence is subsumed by repetitive, rote execution: high-level play means that the set of possible high-score trajectories overlap and converge into a narrow trajectory-space. The game was seemingly not designed to accommodate play at this level; design that once surprised is revealed as rigidly mechanical.

Reflection and Absorption

CE addresses the issue of *Pac-Man's* narrowing trajectory space in several ways, some obvious and some subtle: There are no more level changes, the maze's layout changes throughout. The player is given control over the maze's design, able to change its two sides' designs independently, allowing a wide degree of choice. Pac-Man turns with more granularity, resulting in more variable overall length traveled in a given time. Speed changes are made continuously. All of these choices reduce the apparent determinism of the game, expand its possibility space, clarify and increase the relationships between player interactions and game behavior, and most importantly, enhance the player's sense of the importance of their choices. Because Pac-Man's movement affects much more, much more frequently, and because those effects are not absorbed by *CE*'s systems, the player is made to reason about their decisions over different time-scales. This tight relationship between player choice and clear movement through a large, diverse possibility space is systemic reverberation, conveying emergent gameplay.

Increasing Granularity of Player Interaction Outcomes

CE's cornering increases the window of time for inputting direction on each end of its intersections, allowing players more opportunity to make a turn before and after passing the intersection (see Figure 8). Each step before and after the intersection slows or speeds Pac-Man according to the distance and facing direction. Widening this window and signaling what is happening using feedback increases the number of possible outcomes at every intersection in a way that is imperceptibly granular, still purely deterministic, and now perceptibly signaled with the visual feedback. This expands *CE*'s possibility space and increases the reverberation of the movement mechanic: There is much more that can happen, and the player can see precisely how that is related to movement.

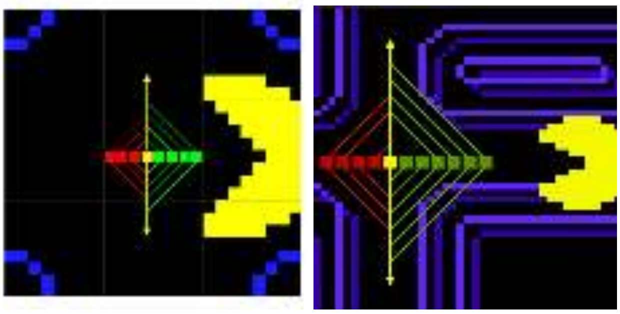

Figure 8: Cornering in Pac-Man (left) (Pittman 2015) and Pac-Man Championship Edition (right), with pre-turn pixels in green and post-turn pixels in red. Each of these corner frames adds or removes a frame of speed to Pac-Man's turn.

Reflecting Outcomes more Frequently and at Different Timescales

One of several ambiguities in *Pac-Man* is what exactly triggers the fruit targets' appearance. Fruit targets appear in response to dot-eating milestones, but nothing in the game indicates this connection. To a novice, the fruit target appears and disappears randomly. To a more experienced player, the fruit target appears early in the level and then later. Players who closely monitor the game, or who study how the game works at a technical level can understand this connection, but for many players it is not a part of their mental model of the game. **For casual players, there is less reverberation coming from the movement mechanic's relationship with fruit target appearance.**

Figure 9: In *Pac-Man Championship Edition*, fruit targets now appear carrying visual feedback (along with a jingling sound effect), calling the player's attention and demonstrating the connection between eating and fruit target appearance.

CE makes the connection between dots eaten and fruit target appearance explicit: When Pac-Man finishes eating all the dots on one side of the screen, a fruit target appears on the opposite side with a jingling sound effect and a sparkling visual effect (see Figure 9). This change in feedback makes the relationship between dots eaten and fruit target appearance more apparent to the player, allowing Pac-Man's movement's reverberation to communicate more of the topology of the game's possibility space.

Increasing Player Relationships to Other Parts

By relating fruit targets and level layout, *CE* exponentially increases Pac-Man's possibility space. As each side of the level is iterated separately, ninety permutations of level layout are made possible, compared to *Pac-Man*'s one. Not only that, but by secondarily relating Pac-Man and the level layout through fruit targets, and by making these relationships apparent through feedback, *CE* demonstrates that traversal into the unique states populating this complex possibility space is visibly and audibly governed by the movement mechanic. The combination of different player-dictated maze configurations create different tensions across the whole maze: One side may have relatively few dots and many power cookies, while the other side may have the inverse. Different configurations may populate the tunnel with dots, forcing Pac-Man to wrap to the other side using the tunnel, while others may not. All of this forces movement to reverberate into level layout, and for that reverberation to describe a necessarily larger, more diverse space.

Allowing State to Accumulate instead of Resetting

In *Pac-Man,* once all dots in a level have been eaten, the level ends: Everything stops and a new level appears, the only non-deterministic attributes carrying forward being points and lives. Points absorb many of the details of previous inputs, comprising a combination of dots eaten, fruit targets eaten, and ghosts eaten (along with combo at time of eating), with much potential overlap. Lives remaining similarly absorb previous input into a combination of points and deaths, with greater potential overlap.

This progression through repeated, disconnected levels represents a deterministic, linear, cumulative absorption of input, where potential sequences of interactions that could have led to a given point have greater overlap as time goes on. The level change is *Pac-Man's* kick-out hole; just as the kick-out hole absorbs momentum, so too do *Pac-Man*'s level changes. *CE* replaces these level changes with a strict five minute timer and constantly-changing level design (see figure 10), allowing player's movement to continually reflect through the space for the entire five minutes, where every movement outcome affects every other movement decision.

Figure 10: Abstract, simplistic representations of *Pac-Man*'s (left) and *CE*'s (right) possibility spaces, and player trajectory through them. Where *Pac-Man*'s levels compress to an ending and restart to a set design, *CE*'s level continuously evolves into a diverse array of possible configurations.

DESIGNING EMERGENCE WITH SYSTEMIC REVERBERATION

Consider Pac-Man's movement to be the player's only form of interaction with the game. When we analyze *Pac-Man's* and *CE's* systemic reverberation, we are analyzing the way that movement (all player interaction) interacts with the game's possibility space. *Pac-Man Championship Edition* conveys a sense of systemic reverberation using Pac-Man's movement (player interaction) by:

1. Communicating the systemic effects of movement more clearly to the player, and more frequently
2. Reflecting movement with a greater degree of granularity
3. Affecting other parts and attributes more with movement, thereby giving the player more ability to control different aspects of game state and navigate the state-space
4. Affecting other parts and attributes *less* with non-movement interaction, further enhancing the player's ability to read the state-space
5. Not resetting levels, thereby allowing a complex state resultant from player movement (ghost position, level design, etc.) to persist and reflect

Drawing from this comparative analysis, we identify several practical game design techniques to increase the sense of systemic reverberation caused by players' mechanical interactions. These strategies encourage growing and diversifying a game's possibility space while making the relationship between player interactions and player trajectory through the space more apparent. These strategies afford the design of emergent gameplay, enhance the sense that players' choices "matter," and can help you effectively communicate complex possibility spaces:

Reflect outcomes more frequently, and at different timescales

By giving the player a more continuous or frequent response to certain mechanical interactions, you give them the ability to "read" the game more closely and better understand how their interactions relate to the game.

By having certain responses happen immediately and others later, you can cause the player to consider both the immediate and distant payoffs of mechanics. This gives the player more ability to choose, and invites them to prioritize.

Think about how your game produces feedback for the player. How frequently can the player see/hear/feel how different parts of the game react to a mechanic? Diversifying these rates of response can force the player to consider a mechanic's effects at different timescales.

Increase the number of possible outcomes produced by player interactions

When mechanical interactions are able to create a wider variety of effects, the state transitions that they cause are able to communicate more precisely in response to player interaction.

When a game responds with greater specificity to the details of mechanical interactions, the game is also communicating to the player that those details matter.

Consider which aspects of mechanical interactions your game might not be paying attention to: time, place, frequency, rhythm, distance to another part, etc. Finding underutilized ways to interpret mechanical interaction gives a designer more tools to design responses.

Increase player relationships to non-player parts with interactions

Connecting player behaviors to more parts of the game gives the player more ability to traverse the possibility space in expressive ways.

These connections, even when subtle, can mitigate the feeling of random chaos. Giving the player secondary or tertiary ways to influence or control random/chaotic elements enhances the sense that their choices matter.

Allowing the player distant, infrequent, or roundabout control over other parts of the game gives them more ability to think ahead and formulate plans, as well as encourage them to test limits and find boundaries of the possibility space.

Consider which parts of your game don't take input from the player in any way, and think of ways to connect the player to those parts, even/especially indirectly. If there is a timer, how might they speed the timer up, slow it down, pause it or start it? Giving tools like these to the player increases the breadth of possibilities in your game, potentially allowing a wider range of strategies and approaches.

Reduce the effect of non-player interactions on attributes affected by player interactions

If a player mechanically affects a given attribute (some part of game state), consider what other behaviors are able to affect that attribute. If the state changes caused by mechanical interactions are reset or overwritten by some other behavior, this absorbs the player's behavior and stifles their ability to read using their mechanics.

Rather than resetting attributes affected by the player, consider changes which still take into account their mechanics. These types of changes allow a game to push back on the player while mitigating absorption.

If a behavior affects an attribute the player also affects, consider having that behavior also be somehow informed by player mechanics. Better yet, if the behavior is affected by a different player behavior than the attribute is, you can build illustrative tension between two behaviors through distant connections between different parts.

Extend the cumulative buildup of outcomes produced through player interaction

When you avoid resetting attributes affected by mechanics, you allow the player to build sequences of events that become evident in attributes' states. This cumulative state stresses the importance of those mechanics to the player.

Allowing mechanical outcomes to accumulate illustrates to the player how the game responds over time, inviting them to consider how their past actions brought them here, and how their current actions might play out. Allowing attributes which accumulate player behavior to interact with and affect one another introduces persistent tension, and gives the player the ability to navigate the possibility space using a long-term balancing act of mechanical interaction.

CONCLUSION

This paper reviews existing literature on game systems design to introduce a term: *systemic reverberation*. Systemic reverberation is an effect which illustrates the process by which game systems and attendant feedback describe a possibility space to player(s) in response to their mechanical interactions. These effects allow players to "echo-locate" through the many-dimensional possibility space generated by a game's system(s). The presence of these effects necessarily imply emergent gameplay. We carry out a formal analysis on *Pac-Man* and *Pac-Man Championship Edition*, similar games with slight design differences which lead to highly contrasting senses of systemic reverberation. By analyzing gameplay through the lens of systemic reverberation, we have developed design strategies which encourage emergent gameplay. Future work could lead to further delineation/taxonomizing of how specific game design decisions foster and communicate emergent, systemic gameplay.

REFERENCES

Adams, Ernest and Dormans, Joris. 2012. *Game Mechanics: Advanced Game Design*. New Riders, San Francisco, CA.

Coke774 and M2. Pac-Man Championship Edition. Game [NES]. (June 2020) BANDAI NAMCO, Tokyo, Japan.

Crawford, Chris and Crawford, Linda L. 1984. *The Art of Computer Game Design: Reflections of a Master Game Designer.* McGraw-Hill Osborne, United States.

Fijak, Marta. 2021. Ghost in the Machine: Authorial Voice in System Design, 'Frostpunk' & Beyond. Video. (19 October 2021). Retrieved February 21, 2024 from https://www.youtube.com/watch?v=e7p4GPqpf18

Internet Pinball Machine Database, The. 2000. Glossary. (2000). Retrieved February 21, 2024 from https://www.ipdb.org/glossary.php

Iwatani, Toru. 2017. Toru Iwatani - Mechanics of Pac Man. Video. (11 January 2017). Retrieved February 21, 2024 from https://www.youtube.com/watch?v=2rck2DWvons

Juul, Jesper. 2002. The Open and the Closed: Games of Emergence and Games of Progression. In *Computer Games and Digital Cultures Conference Proceedings, 2002, Tampere, Finland.* Tampere University Press, Tampere, Finland, 323-329. https://www.jesperjuul.net/text/openandtheclosed.html

Marshalkowski, Benjamin. 2018. Systemic Game Design, Part 1: What are Systems? Video. (3 August 2018). Retrieved February 21, 2024 from https://www.youtube.com/watch?v=NZc7yGdahkY

Maxis. 1989. SimCity. Game [Amiga]. (February 1989). Maxis, California, USA.

Maxis. 2000. The Sims. Game [PC]. (February 2000). Electronic Arts, California, USA.

Maxis. 2002. The Sims Online. Game [PC]. (December 2002). Electronic Arts, California, USA.

Namco. 1980. Pac-Man. Game [Arcade]. (July 1980). Namco, Tokyo, Japan. Last Played February 2024.

Oriz, Eduardo and Nobbs, Christo. 2021. Systems that create ecosystems: Emergent game design. (December 2021). Retrieved Feb 21, 2024 from https://blog.unity.com/games/systems-that-create-ecosystems-emergent-game-design

Pittman, Jamey. 2015. The Pac-Man Dossier. (August 2015). Retrieved February 21, 2024 from https://pacman.holenet.info/

Salen-Tekinbas, Katie and Zimmerman, Eric. 2003. *Rules of Play: Game Design Fundamentals.* MIT Press, Cambridge, MA.

Sellers, Michael. 2017. *Advanced Game Design: A Systems Approach*. Pearson Education, United Kingdom.

Wardrip-Fruin, Noah. 2020. *How Pac-Man Eats*. MIT Press, Cambridge, MA.

Wright, Will. 2001. Will Wright's Design Plunder. Video. (30 June 2020). Retrieved February 21, 2024 from https://www.youtube.com/watch?v=c91IWh4agzU

Zubek, Robert. 2020. *Elements of Game Design*. MIT Press, Cambridge, MA.