Scoring Integrated Meaningful Play in Story-Driven Games with Q-Learning

Alan Bettis[1], Danielle Newberry[2] and Hector Munoz-Avila[1]
[1]Lehigh University, [2] Middlebury College

Abstract

Integrated meaningful play is the idea that player's choices should have a long-term effect on the game. In this paper we present I-score (for integrated), a scoring function for scoring integrative game play as a function of the game's storylines. The I-scores are in the range [0,1]. In games with I-scores close to one, player's early choices determine the game's ending; choices made later in the game do not change the final ending of the game. In contrast, games with I-scores close to zero, players's choices can change the ending until the very end. Games with scores closer to 0.5 provide a more balanced player choice whereby the game's ending still can be changed despite early decisions, but not so much that the ending could be changed at any point.

Introduction

Some games are famous for their storyline and how the players' choices play out in the final ending of the game. For instance, Bioware's Knights of the Old Republic is well-known for a variety of aspects including how the player's choices throughout the game will result in a variety of endings (and cinematics).

Having multiple endings is not a requirement for a game to be considered to have a great storyline. Many first-person shooters such as Valve's Half-Life are linear, yet its storyline is considered memorable. Nevertheless, the long term effect that player's choice has on the ending of a game is one of the most intriguing aspects of game design. Researchers have introduced evaluative definitions assessing player choices. Salen, Tekinbaş, and Zimmerman (2004) define the integrated component of meaningful play to indicate if the players' choices have long term repercussions on the game. The idea is that by enabling long term repercussions of player's choices in the game, the game empowers the players and makes their choices have more meaning.

In this paper, we propose the I-score (for **i**ntegrated) metric. It scores how much player's choice impacts game ending. In games with a I-score close to 0, the player's can change the ending at any point in the game including at the latter stages. On the other extreme, if the game has multiple endings, but "locks" the player once the initial choice is made, then its I-score is 1. If a game has an I-score close to 0.5 it indicates that the game is balanced between these two extremes providing more choice to the player.

Game's storylines can be represented as directed graphs, where each node is an state in the storyline (e.g., beginning of the game, each of the possible endings) and edges represent transitions between states as a result of player's choices (Riedl and Young 2006). The basis for computing the I-score is the use of Reinforcement Learning (RL) techniques with the aim at finding the "true" value (defined in RL is a metric of the suitability of every state to reach a goal based on the rewards) of every state based on the reward function. We then find the distances between those true state values to compute the overall I-score of the graph.

The premise of our work is that integrated meaningful play assess the relationship between player choices and the game ending. As the value of being in a state is intrinsically linked to player choice, Reinforcement learning finds the value of each state.

We present an empirical evaluation on both synthetic graphs, representing different topologies of player choices, as well as actual games and discuss their resulting I-scores.

Preliminaries

Our work links two different concepts: Q-learning and Meaningful Play. We introduce these concepts here.

In reinforcement learning, the agent aims to maximize its future rewards while interacting with the environment from a starting time t: $r_t + r_{t+1} + ... + r_T$ (Sutton and Barto 2018). For our purposes we focus on episodic tasks, where the agent terminates its execution after reaching a concluding state.

Tabular reinforcement learning algorithms maintain a two dimensional table $Q : S \times A \rightarrow R$, where S represents the states that the agent can visit, A represents the actions that the agent can take, and R represents the real numbers. The entry $Q(s, a)$ indicates the estimated reward the agent gains from taking action a in state s. The estimated value $Q(s, a)$ is updated after the agent takes an action and receives a reward using the following formula:

$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \alpha[r + \gamma \max_{a'} Q(s',a')]$$

Where s' is the state reached after executing action a in state s. α is the step-size parameter determining how much weight to give the previous estimate $Q(s,a)$ and how much weight to give the update (i.e., the $[r + \gamma \max_{a'} Q(s',a')]$ part of the equation). γ is the learning rate indicating how much weight to give to prior estimates of $Q(s',a')$.

Reinforcement learning algorithms can use a variety of techniques such as ϵ-greedy: with a probability $1-\epsilon$ where the agent chooses the greedy action $pi(s)$ and with a probability ϵ it pics a random action. Based on the estimates, $Q(s,a)$, the so-called greedy policy can be defined:

$$\pi(s) = \max_{a'} Q(s,a')$$

To guarantee that future reward is maximized, the policy enforces that for every state s the action a' with the highest current estimate $Q(s,a')$ is chosen.

In *Rules of Play: Game Design Fundamentals* Salen et al. proposes a game design framework for not only computer games, but all games — from board games to sporting games. Meaningful play is highlighted as a core concept of game design and integral to a game's success. Salen et al. define meaningful play as a relationship within a game between action and outcome. There are two types of meaningful play: integrated and discernable. Integrated meaningful play means that a player's action has some long term consequences, i.e. the decision or play is woven or integrated into the game as a whole. Discernable meaningful play refers to when a player has immediate or discernable consequences to a given action. This work posits that successful games include game actions and outcomes that are both discernable and integrated (Salen, Tekinbaş, and Zimmerman 2004).

The notion of meaningful play provides a conceptual assessment of the design of a video game. It evaluates the relationship between actions taken by a player and the outcomes of those actions. It says that in order for the interaction between players' actions and the actions' outcomes to be meaningful, they must be **discernable** and **integrated**.

Discernable means that the immediate outcome of the action is observable by the player. For example, when the player press the "w" key, they observes their avatar moving forward. Integrated means that the players' actions have a long-term effect on the course of the game. For example, in games like Knights of the Old Republic, players actions such as killing or saving a particular NPC, will assign "good" or "bad" karma points and these in turn will determine how the storyline in the game will end.

This work solely focuses on assigning a score to the amount of integrated meaningful play in story driven games. Our algorithm determines the integrated impact of every player choice made using two main processes: (1) We calculate overall "meaningfulness" of a choice; (2) we weigh the obtained choice value to account for integration level. First, to calculate the impact of a choice we use Q-learning to determine how much reward is obtained by making a given choice with the goal of getting to a desired ending state. Then we use the Manhattan distance to compare all possible routes from the same starting point to a desired end point to determine the variance of paths, i.e. the importance of certain choices. Second, we weigh user's earlier choices higher than future ones to account for the fact earlier choices have the opportunity to have a more integrated impact than choices made near the end of a game.

Scoring the Integration Level of a Game

We present a procedure computing the I-score, a scoring function I: $GAME \rightarrow [0,1]$ such that if a score is closer to 0.0, this indicates a high degree of overlap between the paths from the beginning of the game to each ending of the game. Conversely, if a score is 1.0, the paths from the start of the game to each ending have more separation at the end.

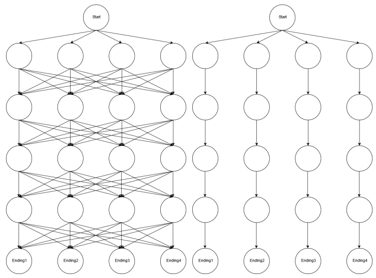
For our study we developed synthetic graphs to represent choice driven games with varying amounts of choices for players. The synthetic topologies have a single starting node connected to four succeeding nodes. There are four levels, and a final ending level each with four nodes. The term "branch" will refer to the direct paths made from the first decision nodes to the respective ending nodes vertical to them.

Figure 1 shows two of these topologies. The starting node represents the starting state in a video game. After a player makes an in-game decision, the game state transitions to the corresponding connected node.

We begin with both extremes. In Figure 1 (a), each node in each level is connected to every subsequent level's node allowing for a transition into any other branch. Most player choices made during the duration of the game have no impact on the ending of the game. Instead, only the final player decision determines the game's ending state. We refer to this topology as "non-integrated" because aside from the last choice, all other decisions made by the player have no affect on the final outcome of the game. Its I-score is given by I(non-integrated) = 0.1585, an unsurprisingly low score. Only the player's final decision has a direct impact on the ending, but this decision is not integrated because it is made at the end of the game.

For our opposite synthetic graph, we construct a topology such that after a player makes their first choice, they become locked into a specific branch with no possibility of changing their outcome. Thus, the very first choice the player makes determines the outcome. We refer to this topology as "rigid-integrated". It is shown in Figure 1 (a). It has an score I(rigid-integrated) = 1.

Underlying assumptions on game structure. As a first work computing a numeric measurement of integrated meaningful play, we make some assumptions about the graphs of the games we have chosen to test. First, we assume that the graph is a directed acyclic graph (DAG). That is, from any game's state there is no sequence of player's choices that will lead to that same state. Many games do have such cycles, but we have chosen games that do not have such cycles for simplicity. Second, we assume that the



(a) (b)

Figure 1: (a) Non-integrated topology; regardless of the choices made the player can always reach any of the endings; it has an I-score of 0.15; (b) Rigid-integrated topology; once the player makes a choice in the first decision, the outcomes are fixed for the remaining of the game; it has an I-score of 1.

graphs have multiple ending states. Games such as Knights of the Old Republic have multiple endings depending on the player choices. Multiple endings are not a requirement for a well-received narrative-based game, but they provide an intuitive way to compare routes for scoring a game.

While many successful narrative-based games only have one ending, or include cycles in their structure, this paper describes a new proof-of-concept procedure. As such, we leave these other types of game structures for future work.

Computing Integration Score

We present a procedure to compute the I-score of a game. It is a composite of four elements:

$$I(Game) = (M \circ N \circ W \circ Q)(Game)$$

Computing the Q-values

For the games we are considering, a story can be seen as a DAG, where nodes are precise points in the story and edges are players' choices advancing the story. Our goal is to analyze the overall level of integration in a story. To do so, we use Q-learning to calculate the reward value of every state-action pair by defining a matrix of dimensions $1 \ldots Nodes \times 1 \ldots Nodes$. Our scoring system accepts this matrix as input, with all cells initialized to 1. Additional parameters include the position of the start node, representing the beginning of the story, and the positions and number of ending nodes, representing all possible story endings. We give a reward of 100 to the desired ending state in order to incentivize our Q-learning agent to reach this destination. We then run Q-learning on the resulting matrix for each ending state. After 100,000 iterations, the resulting Q-tables show the expected rewards for each choice in each given story point. For each ending, we get a separate Q-table with the updated reward values. Let this procedure be defined by the function $Q(Game)$, which accepts an adjacency matrix representation of a game as input and outputs a series of n Q-tables, where n is the number of available endings in the game's story.

Computing Weighting Function W

Integrated play is defined as the effect a given action has later on in a game. Therefore, it stands to reason that earlier actions have more opportunity to have a long-lasting impact on the game. To reflect this, we assign weights to actions based on how early in the game they occur, with more weight given to earlier actions.

We calculate a set of n weights such that they sum to 1, where n is the number of transitions in the longest route in the graph. We then apply these weights in descending order, with the higher weights being applied closer to the graph's start node and smaller weights being closer to the graph's endings.

Algorithm 1 describes the procedure for calculating the set of weights. It first initializes an empty set of weights (Line 2). Then, it calculates the slope of a line according to *graphHeight*, the length of the longest route from start to any ending in the graph (Lines 3-10). Our initial unprocessed weights are given by *i*slope*. Once this is done, the algorithm calculates the average difference, *toAdd*, between the sum of all unprocessed weights and 1 (Line 7). Because the unprocessed weights are much smaller than 1, we add *toAdd* to each *i*slope* (Lines 8-10). Lastly, we reverse the weights to put them into descending order.

Algorithm 1 Weight Calculation

1: **procedure** GETWEIGHTS(graphHeight)
2: new List $weightSet$
3: $slope \leftarrow 1/graphHeight^2$
4: $sum \leftarrow 0$
5: **for** $i \leftarrow 1; i \leq graphHeight; i++$ **do**
6: $sum \leftarrow sum + (i * slope)$
7: $toAdd \leftarrow (1 - sum)/(graphHeight)$
8: **for** $i \leftarrow 1; i \leq graphHeight; i++$ **do**
9: $weight \leftarrow ((i * slope) + toAdd)$
10: $weightSet.append(weight)$
11: $weightSet.reverse()$
12: **return** $weightSet$

Once we have a set of linearly descending weights, we can apply them to the Q-tables to construct a set of weighted Q-tables that we will call "I-tables." This is done by conducting a depth-first search of each Q-table and multiplying each Q-value by the weight corresponding to the Q-value's height in the graph.

If two or more branching paths in the graph of differing lengths re-converge, the "height" of the node could be the height of either prior node plus one. However, this is not consequential in the overall score, as the weights maintain their descending order before the branching and after the re-convergence. Therefore, choices are still weighed higher when found earlier, preserving integration.

Integrated play means that a decision has long-lasting consequences, and a decision made earlier on has more opportunity to have long-lasting impact. Therefore, this function allows our scoring system to detect integration in games by placing higher value on decisions made earlier in the story. Let this procedure be defined by the function $W(Q - tables)$, which accepts a set of n Q-tables as input and outputs a set of n I-tables.

Computing the Normalized Scores N

To ensure that our final score is normalized between 0 and 1, we run softmax on a vector containing all non-zero values in a given I-table, producing a set of normalized values. We can then take our normalized set and overwrite the original values with their normalized values.

In order to normalize all values in a given weighted Q-table, all non-zero values in the table are compiled into a one-dimensional vector. As non-zero values signify connections, they are the only values of interest in this operation. We then use softmax to normalize the values in this vector, and replace the old non-zero values in the table with the corresponding new ones from softmax (Bridle 1990). Let this procedure be defined by the function $N(Q-Tables)$, which accepts a set of n weighted Q-tables as input and outputs a set of n normalized, weighted Q-tables.

Computing the Manhattan Scores M

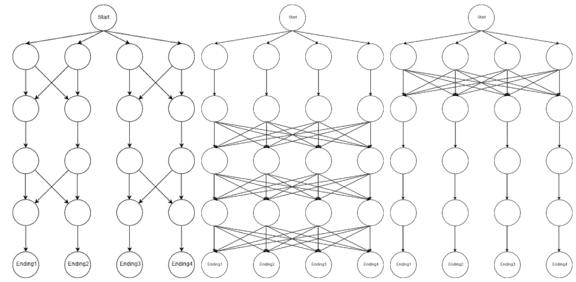
Once this is done for each table, we can calculate the Manhattan distances in a pairwise fashion between each I-table, getting a sense of how different each route is in the process. Once these distances are averaged, we have our I-score.

Finally, we take the Manhattan difference between each pair of the game's I-tables. If two I-tables for different endings have different paths, then the Manhattan distance will be higher, reflecting the greater amount of integrated play as a result of the game. Conversely, if a nearly identical path can be taken to two different endings, the Manhattan distance will be lower. The final average is divided by 2 so that a different choice between two I-tables is not counted twice. Let this Manhattan distance procedure be defined by a function $M(Q - tables)$, which accepts n weighted, normalized I-tables as input and outputs a numeric score from 0 to 1.

Empirical Evaluation

Synthetic Graphs

In addition to non-integrated and the rigid-integrated graphs we created other synthetic graphs as shown in Figure 2. They combine various levels of inter-connectivity.



(a) (b) (c)

Figure 2: (a) The two-on-two graph allows for crossover in every other level alternating between the two endings.; it has an I-score of 0.9708; (b) The rigid-top topology, top half is identical to the rigid-integrated and the bottom half is identical non-integrated graph; it has an I-score of 0.1590; (c) The rigid-bottom topology, inverts the connectivity of the rigid-top. It has a score of 0.9995

Game Graphs

We choose four story-driven video games on which to test our scoring system because we were able to formulate their topologies from online sources. For each game, we construct a simplified graph where each choice is a transition from one node to another. Choices related solely to non-story elements of the game such as achievements are ignored for simplicity. From these graphs, we construct adjacency matrices to pass as input to our scoring algorithm.

The first and most complex game that we inspect is *Steins;Gate*, a game of the visual novel genre. [1] Game play primarily takes the form of dialogue trees where the player makes decisions from time to time when prompted. The player can check the protagonist's cell phone at any time, and respond to messages from other characters using one of a finite number of options. For simplicity, dialogue routes that only lead to achievements are ignored.

The graph constructed from Steins;Gate is shown in Figure 3. There are many branching points in the story with 6 endings of varying difficulty to achieve, the true ending being the most difficult. Each time a critical decision is made, the player is denied access to one ending, but can still access all the remaining endings from the new branch. Steins;Gate received an I score of 0.6829.

Next, we investigate a browser-based puzzle game called *No One Has to Die*.[2] In each level of this game, the player must manipulate elements on the board to save characters from a fire, but the puzzles are always designed such that one character must be sacrificed. The ending of the game is dependent upon which characters remain alive. Once all endings have been seen, the "true route" is unlocked, which is ignored as it is linear, and, thus not of interest for this study.

No One Has to Die's graph is much simpler than Steins;Gate, with 5 major endings based on who lives or dies. There are 3 choices the player will make in a given play through of the game. The first choice forces the player down one of two paths, each of which can either access one of a different pair of endings, or rejoin on a single ending. This game was assign a I-score of 0.6782.

In the game *Shadow the Hedgehog*,[3] players move through levels collecting rings and engaging in combat with enemies much as they would in the *Sonic the Hedgehog* series. The key difference from typical Sonic the Hedgehog games is that each level in Shadow the Hedgehog contains multiple objectives, and which objective the player completes leads them to a different next stage. Based on which

[1]https://en.wikipedia.org/wiki/Steins;Gate

[2]https://levelskip.com/puzzle/No-one-has-to-die-Walkthrough

[3]https://en.wikipedia.org/wiki/Shadow_the_Hedgehog_(video_game)

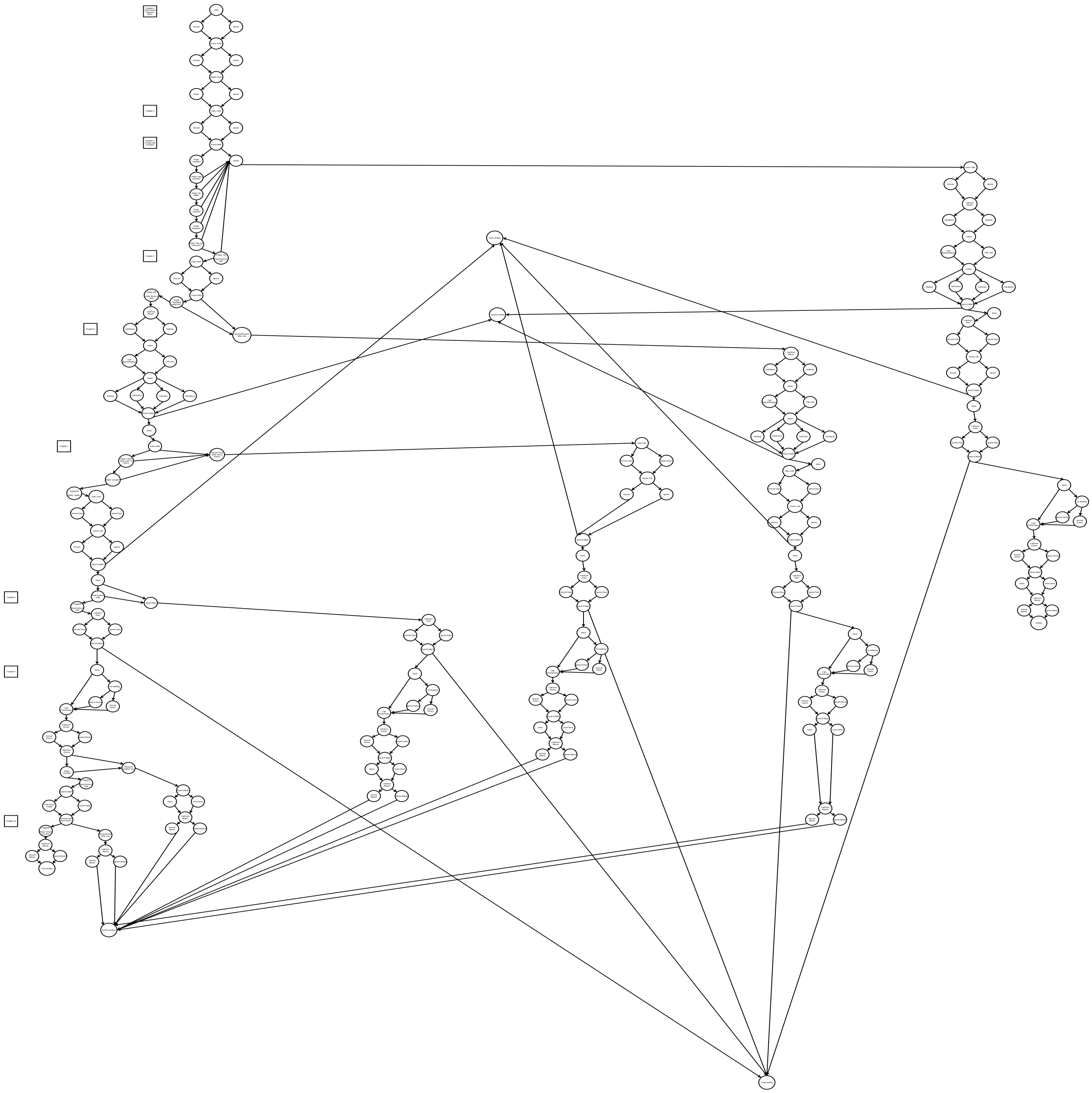


Figure 3: Graph of the Steins;Gate game.

route the player completed, they will receive one of 10 different endings. Similarly to No One Has to Die, there is a “true ending” route unlocked once all other endings have been seen, but this is once again linear and ignored for the purposes of this paper. The graph of Shadow the Hedgehog is constructed with each node representing a level, and each edge representing one of the possible objectives to complete. This game is notably non-committal in comparison to the other games we have discussed so far, allowing the player to move between several routes almost freely, and only denying access to certain endings after the third or fourth choice. Shadow the Hedgehog received an I-score of 0.8354. This score closer to 1 is indicative of higher difference between routes and more “strictness.” While there is less impact on the player’s early choices, the player decisions begin to matter more and more as they reach the end.

The last game we investigate is *Zero Escape: Virtue's Last Reward*,[4]the second installment in the "Zero Escape" series of games. Zero Escape is a series of visual novel games in which the player solves puzzles to advance the story, and occasionally makes decisions that change the route. Among the games we cover in this paper, this game is unique in that it prohibits access to certain endings at first through various means. Information required to solve certain puzzles is found on different routes, making progression essentially impossible without having been down the correct routes. Other endings are fully locked, the player being met with a screen reading "To be Continued..." if they reach the ending prematurely. This is meant to enforce a degree of linearity to the storytelling.

For simplicity, we construct two different graphs: A "complete" graph with prologue and all endings "unlocked", and a "locked" graph with no prologue segment and only initially unlocked endings. Both graphs have a very similar structure to that of No One Has to Die, albeit with more endings and length.

The complete graph has 20 accessible endings, with the first decision the player makes denying access to over half of the endings. With the large amount of denial from endings, the complete graph receives a I-score of 0.9763. The similarity to the synthetic rigid-integrated graph is expected, as there is no overlap between routes in this game.

Meanwhile, the locked graph has a nearly identical structure, but only allows access to 9 endings. This proves to have very little impact on the I-score, as this graph receives a score of 0.9983. The increase in score can be attributed to the reduced number of endings, meaning that each choice denies a player access to a higher percentage of the total number of endings.

Experimental Setup

Our Q-learning parameters are as follows: $\alpha = 0.90$, $\gamma = 0.75$, and $\epsilon = 0.20$. For each game, we run Q-learning on the game's weighted adjacency matrix 100,00 times. We repeat this process 10 times and average the outputs.

Experimental Results

Table 1 shows the I-scores; the synthetic graph's I-scores are very near to either 0 or 1– the ends of the spectrum. Both the non-integrated and the rigid-top graphs receive very low scores demonstrating that graphs with lots of crossover between branches thereby the choices have little impact. In fact, the non-integrated graph allows for complete crossover – a player in states can always move to any state in the following level of the topology, yielding an I-score near 0. On the other hand, the rigid-integrated, rigid-bottom, and two-on-two graphs have very high scores. This illustrates the limited amount of crossover between branches in their topologies. The rigid-integrated graph receives a perfect score of 1 indicating that there is absolutely no crossover between branches.

Game	Score
Steins;Gate	0.6829
No One Has to Die	0.6782
Shadow the Hedgehog	0.8354
Zero Escape: Virtue's Last Reward (Unlocked)	0.9763
Zero Escape: Virtue's Last Reward (Locked)	0.9983
Non-Integrated	0.1585
Rigid-Integrated	1.0000
Rigid-Top	0.1590
Rigid-Bottom	0.9995
Two-on-Two	0.9708

Table 1: Resulting I-scores

Both unlocked and locked Zero Escape: Virtue's Last Reward yields I-scores in the range of 0.9 indicating a close to rigid topology. Shadow the Hedgehog has a slightly lower score of 0.8354 suggesting that this game has slightly more divergence between the branches of its topology. In a preview of Shadow the Hedgehog, Gooseberry commented "gamers will decide in a race against time by choosing between any number of routes by which to unveil the secrets of his past. Multiple endings will reflect critical decisions at every stage of the game. Gamers have ultimate control over Shadow's past, with the potential to rewrite his history time and time again." [5] The game's high score reflects the gamer's "ultimate control" early game decisions do not immediately lock player's out of endings, late decisions made in the game still have the power to swap branches.

Steins;Gate and No One Has to Die have similar scores in the range of 0.6. Both of these games have complex topologies that contain a mixture of paths. We hypothesize that games between these two extremes, with a I-score close to 0.5 have more interesting story lines. Daily Star reviewers of Steins;Gate remarked, "As you proceed, once trivial choices you make reveal themselves to have a direct effect on the game's finale. Multiple endings both good and bad are up for grabs hindering even seemingly simple decisions with a heavy weight, even more so as you debate the consequences of altering time and the potential butterfly effect it'll have on the characters you've fabricated deep ties with."[6] This quote exhibits how meaningful play is highly impactful on player experience. Even trivial choices end up having an effect on the ending. When a player must put thought into their game decisions it can make for a more engaging, thought provoking, interactive game experience. A review of No One Has to Die posits, "The main highlight is the plot and the music. Both had me intrigued, seriously absorbing. I always love it when alternate paths somehow link together. It is definitely worth finding all the endings!" [7] This quote emphasises the converging game paths. The reviewer suggests that players' choices creates a more enjoyable game experience.

[4]https://en.wikipedia.org/wiki/Zero_Escape:_Virtue's_Last_Reward

[5]https://cheatcc.com/xb/rev/shadowhedgehogreview.html

[6]https://www.dailystar.co.uk/tech/reviews/steins-gate-elite-ps4-review-16905592

[7]http://gamescheat22.blogspot.com/2013/04/no-one-has-to-die-game-review.html

Related Work

Player enjoyment is difficult aspect of a game to quantify, thus there is a need for a validated and generalized model that can be used to design and evaluate game enjoyment (Bostan and Öğüt 2009; Sweetser and Wyeth 2005).

Fabricatore, Nussbaum, and Rosas (2002) used single player games to construct a qualitative design model aiming at understanding what players want in a video game. Participants had a playing session with one of 39 selected games (picked by their popularity). During the play session participant's activities and opinions were logged. This study resulted in the formation of a hierarchical design model using entities, scenario, and goals as three aspects of the game that were found to be deterministic of the player experience.

In Desurvire, Caplan, and Toth (2004) a study was conducted to assess the usefulness and validity of various heuristics by tracking participants navigation and feedback in a game shell. The user study provided evidence to using heuristics to evaluate games for playability. They determined game heuristic categories such as game story (e.g., character development) and game play (e.g., challenges users must overcome to move forward throughout a game).

Sweetser and Wyeth (2005) report on a model to evaluate player enjoyment. Various design heuristics proposed in previous research are integrated into a model to evaluate game enjoyment using flow. Flow occurs when the challenge level matches the skill level of the user. The proposed model GameFlow adapts the idea of flow experiences to games. The model consists of components such as: concentration, challenge and skills. Bostan and Öğüt (2009) further develop a GameFlow model specifically for RPGs to aid in maximizing player experience. User's freedom of choice and meaningful play are deemed important aspects of a games storyline. A game's difficulty curve should keep player engagement high and minimize frustration by increasing game difficulty as the player progresses through the game.

Purdy et al. (2018) study a collection of story features to assess an storyline's quality. For instance, the temporal ordering feature looks at the plausibility that one sentence follows from its previous sentence in the story. Our study is orthogonal to story's quality. We are looking at measuring the integrated level of a game between the player's choices and the outcome of the game.

Mawhorter, Mateas, and Wardrip-Fruin (2015) present a system capable of generating choices to achieve poetic effects. The system assumes that the player has intrinsic goals such as avoiding injury and diffuse threats to innocent NPCs. It can generate three kinds of choices in relation to the player's goals. For instance, in dilemma choices, player's choices will lead to achieve one of the player's goals but will cause other goals to fail. The focus of these criteria is on individual players' choices whereas our focus is on how the player's choices affect the outcome of the game.

Conclusions

We introduce the I-score, a metric that evaluates the integrated level of a game with respect to its possible endings. This is the first work attempting to provide a numerical measure integrated meaningful play for storylines. As such, we made some underlying assumptions about the kinds of graphs in the storylines: (1) we assume there are no loops in the storyline: that is, there is a sequence of player's choices from a state decisions that will lead to that same state; (2) we assume that games have multiple ending states. While some games do violate these assumptions, for the purposes of this paper we are only considering games that do not. Our empirical results show that games that "lock" the player on early choices have a I-score close to 1 whereas those where choices do not matter as much have scores closer to 0. Games that contain a balance of the two have I-scores close to 0.5. In future work, we would like to examine games such as Bioware's Mass Effect 3, where endings are also influenced by internal scoring functions. Our code for this paper is available at https://github.com/abettis56/meaningful-play-ml.

Acknowledgements. This research was supported by NSF REU grant 1757787, NSF grant 1909879, ONR grants N00014-18-1-2009 and N68335-18-C-4027. Any opinion, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the funding agencies.

References

Bostan, B., and Öğüt, S. 2009. Game challenges and difficulty levels: lessons learned from rpgs. In *International simulation and gaming association conference*, 1–11.

Bridle, J. S. 1990. Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. In Soulié, F. F., and Hérault, J., eds., *Neurocomputing*, 227–236. Berlin, Heidelberg: Springer Berlin Heidelberg.

Desurvire, H.; Caplan, M.; and Toth, J. A. 2004. Using heuristics to evaluate the playability of games. In *CHI'04 extended abstracts on Human factors in computing systems*, 1509–1512.

Fabricatore, C.; Nussbaum, M.; and Rosas, R. 2002. Playability in action videogames: A qualitative design model. *Human-Computer Interaction* 17(4):311–368.

Mawhorter, P. A.; Mateas, M.; and Wardrip-Fruin, N. 2015. Generating relaxed, obvious, and dilemma choices with dunyazad. In *Eleventh Artificial Intelligence and Interactive Digital Entertainment Conference*.

Purdy, C.; Wang, X.; He, L.; and Riedl, M. 2018. Predicting generated story quality with quantitative measures. In *AIIDE*, 95–101.

Riedl, M. O., and Young, R. M. 2006. From linear story generation to branching story graphs. *IEEE Computer Graphics and Applications* 26(3):23–31.

Salen, K.; Tekinbaş, K. S.; and Zimmerman, E. 2004. *Rules of play: Game design fundamentals*. MIT press.

Sutton, R. S., and Barto, A. G. 2018. *Reinforcement learning: An introduction*. MIT press.

Sweetser, P., and Wyeth, P. 2005. Gameflow: a model for evaluating player enjoyment in games. *Computers in Entertainment (CIE)* 3(3):3–3.