Generative Agents for Player Decision Modeling in Games

Christoffer Holmgård, Antonios Liapis, Julian Togelius
Center for Computer Games Research
IT University of Copenhagen
Copenhagen, Denmark
{holmgard,anli,juto}@itu.dk

Georgios N. Yannakakis
Institute of Digital Games
University of Malta
Msida, Malta
georgios.yannakakis@um.edu.mt

ABSTRACT

This paper presents a method for modeling player decision making through the use of agents as AI-driven personas. The paper argues that artificial agents, as generative player models, have properties that allow them to be used as psychometrically valid, abstract simulations of a human player's internal decision making processes. Such agents can then be used to interpret human decision making, as personas and playtesting tools in the game design process, as baselines for adapting agents to mimic classes of human players, or as believable, human-like opponents. This argument is explored in a crowdsourced decision making experiment, in which the decisions of human players are recorded in a small-scale dungeon themed puzzle game. Human decisions are compared to the decisions of a number of a priori defined "archetypical" agent-personas, and the humans are characterized by their likeness to or divergence from these. Essentially, at each step the action of the human is compared to what actions a number of reinforcement-learned agents would have taken in the same situation, where each agent is trained using a different reward scheme. Finally, extensions are outlined for adapting the agents to represent sub-classes found in the human decision making traces.

Categories and Subject Descriptors

I.2.1 [**Artificial Intelligence**]: Applications and Expert Systems

General Terms

artificial intelligence, game design, decision making

1. INTRODUCTION

This paper describes an approach to modeling, grouping and interpreting players based on their inferred utility function modeled through reinforcement learning in the form of a generative agent. It proposes how this could be extended to adapt to derived player groups through a process of clustering and inverse reinforcement learning. The example presented here uses trained Q-learning agents with manually configured reward parameters as a priori defined personas, but in principle any agent with a trainable, configurable reward function could be used to generate easily interpretable generative player models for player characterization and design support.

Player modeling in its various expressions facilitates at least one of four purposes: the description, prediction, interpretation, and in some cases reproduction of player behavior [9, 14]. All four purposes are rarely addressed simultaneously in the same model for theoretical and practical reasons. An interest in understanding groupings of players might not necessarily entail an interest in accurately *predicting* or *reproducing* their behavior. Inversely, in order to create a player model that reproduces player behavior, it might not be necessary to account for *why* players exhibit a certain behavior in-game — only that a reproduction is convincing to a human spectator. Still, certain areas of investigation or specific applications might mandate the pursuit of player models that address all four purposes at once.

This paper expands on previous work [5] and attempts to span all four purposes outlined above: it aims to describe, predict, and reproduce player decision making with the overarching goal of facilitating its interpretation. By modeling players as decision making agents and using these models to characterize players by their induced motivations, a high-level sketch of the players' decision making processes is drawn facilitating the interpretation of their preferences. The approach draws on the theoretical framework of prospect and decision theory, considering every action in a game a decision made under uncertainty [11, 8].

This approach is based on three fundamental assumptions: The first assumption is that players exhibit a particular decision making tendency or style when playing a particular level or game, and that this tendency can be captured and expressed by approximating a *utility function* that shapes their decisions in-game. This utility function is *latent* in the sense that it cannot be observed directly. The second assumption is that in order to *validly* model this assumed utility function, the elements and procedures of the utility function, as a psychological construct, should be explicated in a manner that accounts for the key components and processes of the player's psychology, the outcomes of which can be empirically observed. The third assumption is that a priori outlined models of player decision making styles can be used as archetypes or personas by comparing their genera-

tive output with the empirical observations of human decision and that a well-motivated artificial process that generatively mimics a player constitutes a valid abstract model of the player's internal process.

In the following section, we will first discuss related work from psychology and artificial intelligence, and the epistemological assumptions in that work. In Section 3 we explain the general structure of our experiments as well as the artificial intelligence methods involved. Sections 4 and 5 describe our method for data collection from human players, and the results of our attempts to classify human playtraces according to agreement with generative agents. We conclude by discussing the potential and limitations of the current work and overall methodology.

2. RELATED WORK

The presented method of player decision style modeling draws in parallel on the literatures on decision theory, psychometric validity, and player modeling. This section outlines the insights drawn from each field and motivates the synthesis of the three.

2.1 Decision Theory

Psychology, behavioral economics and game theory share a common history under the umbrella of decision theory which tries to describe decision making through formal models. One of the central ideas of decision theory is that any decision a human makes under uncertainty, due to incomplete information or a stochastic outcome, is guided by a *utility function* that determines the decision makers willingness to take risks for an expected reward.

Utility to the decision maker is considered idiosyncratic, and decision theory makes no general claims about this, but typically defines a particular conception of utility a priori. Whatever is desirable to the decision maker is a potential source of utility to various degrees. It prescribes that, given its conception of utility, an agent acts rationally when it optimizes, within its computational constraints, its actions to achieve it [8]. The utility function describes the decision maker's risk/reward policy for this optimization.

Here, the purpose is to develop a method for decision modeling that is relevant for a wide range of computer games, including ones that support (even if they might not suggest) unstructured play in the game world — or have competing or even conflicting goals. For that reason it is most relevant to interpret any player input to the game as a decision that is expressive of a utility function that is shaped both by the interaction between the game's overt rules, its expressive space in total, and the player's motivations and capabilities. Any action the game *affords* the player [4] becomes a potential source of utility. If any action in the game can be taken as an output of the player's utility function, this in turn allows for inducing the player's concept(s) of utility by approximating and then interpreting the utility function.

Since the utility function weighs the values of the constituents of future game states, relative to the risk involved with potentially attaining them, it is necessary to define these constituents before attempting to model the utility function — constructing a selection of affordances that *could* provide utility in the game. This comes with the risk of identifying only a subset of the actual affordances or perhaps picking the wrong ones altogether.

Once an acceptably large set of possible affordances are defined, an approximation of the player's utility function could technically be accomplished by any generative computational method capable of simulating the human decision maker, rule-based or search-based. However, since the interest here is not only reproducing the utility function, but also interpreting the computational generation of a given utility value as an abstract representation of the player's same process, it is necessary to apply methods that allow for the inspection and interpretation of the weightings of the affordances behind the utility function. Once successfully constructed, such a model is then interpreted as an abstract simulation of the player's decision making process. The following section briefly argues why this methodological approach can be considered appropriate in terms of psychometric validity.

2.2 Validity in Latent Trait Modeling

To construct player models that aim to discriminate between players or predict their actions, by modeling a process that is completely internal to the psychology of the player and therefore unobservable, an argumentation for the validity of the proposed model of the player's psychology is necessary. The work presented here attempts to induce the player's sources of utility, treating the utility function as a latent trait or state within the player, motivating her behavior. Recent research in psychometrics argues that a particular test or model for measuring a latent attribute is valid if "a) the attribute exists and b) variations in the attribute causally produce variations in the measurement outcome." [2]. Although at first glance this seems intuitive, the necessity of a causal relation between the attribute and the measurement outcomes puts an explanatory onus on the theoretical framework and assumptions of the model. A psychological concept that cannot produce theoretical reasons for assuming the modeled processes in the psychology of the player runs the risk of regressing to operationalism where the process in the player is defined as what is measured through the empirical methods [7], potentially mistaking outcome correspondence for process correspondence. To avoid this risk, a model that claims to represent unobservable processes in the psychology of a player needs a clear mechanistic chain of inference from the context to the player action to facilitate description, prediction, interpretation, and reproduction. Otherwise, it cannot claim to model the internal process of the player, but only produces a potentially unrelated, even if effective, mapping between the input and output states [1].

This is specifically what this work attempts to address by developing a model of player decision making that takes into account high-level characteristics of the human decision process, while remaining reasonably intelligible, by making strong claims about what aspects of a decision problem are evaluated by the player and what importance the player attributes to each aspect in the form of a persona.

2.3 Player Modeling

Yannakakis et al. [14] present a high-level overview of player modeling approaches and argue that player models always, at least in an abstract sense, incorporate the whole player either overtly or tacitly. The paper usefully separates model-based and model-free player modeling approaches, while pointing to the fertile, hybrid middle ground between the two. From this perspective, the approach taken here

is model-based in the sense that it makes strong assumptions about the psychology of the player and represents it in the form of agent-personas, but the actual agent training is model-free in the sense that a Q-learning agent is used. As such, the method presented here is a hybrid one.

Smith et al. [9], present a useful, inclusive taxonomy of player models, identifying opportunities for filling gaps in the already known gallery of approaches to player modeling. They present four facets of player models that can be used to describe their *kind*: the scope, purpose, domain and source of the player model. The method that is presented here would, under their taxonomy, be categorized as a Class Induced Generative Action model. Smith et al. specifically note that "Class models are more difficult to motivate in an academic context, requiring either justification of a theory of stereotypes or aggregation of sufficient individual data to build up class descriptors. Thus, we expect class models to be used more in practice than they are reported.". This precisely touches upon the considerations of validity outlined in the preceding section, and helps explain why the class based category of academic player models has no examples in Smith et al.'s survey. The quote also describes the potential applicability of class based models: Stereotypical players, or personas, are widely used in game design and development for guiding content creation [12, 3], taking the place of play testers when actual play testing is infeasible or undesirable. Typically, a game designer uses the persona as a starting point for imagining what the persona would do in a particular part of the game, or actually plays the game while informally simulating the persona's play style. This implies that the game designer has a mental model of the decision making process of the persona, typically based on previous experience and the interpretation of qualitative and/or quantitative data from play testing, metrics, etc. The purpose of our modeling method is obviously not to supplant this part of the game design process, but to provide the game designer with an external representation of not only *how* different personas would play the game, but at the abstract level also *why*. Such a model could form a point of comparison and contrast to the game designer's internal mental model or become part of a mixed-initiative content authoring tool, suggesting content suitable for one or more personas, configured by the designer, adapted to human data, or built as a a hybrid of the two.

3. TESTBED

For the purpose of exploring the argument presented above, a simple testbed game was created along with a set of archetypical generative agents.

3.1 Game Environment

The game environment, MiniDungeons (see Fig. 1), aims to evoke the fundamental mechanics of a rogue-like dungeon exploration game. It puts the player in a two-dimensional dungeon on a grid of 12 by 12 tiles, viewed from a top-down perspective. Tiles are either passable or impassable to the player. Passable tiles may be occupied by monsters, rewards, potions, the dungeon entrance or the dungeon exit. All tiles and their current state is visible to the player, so the game applies no notion of fog-of-war or limited visibility. The player has a hitpoint counter and a treasure counter, and the player loses the level if her hitpoints (HP) drop to zero. The player starts each level at the dungeon entrance with

Figure 1: The game environment on one of the levels used in the experimental protocol. The hero, shown in gray armor, moves around the level collecting treasures (brown closed chests), potions (red bottles) and killing enemies (green goblins). The hero starts at the entrance (stairway leading up, left of the screen) and the level ends when the exit is reached (stairway going down, right of the screen). The hero's hitpoints are shown at the bottom, along with the number of treasures collected and the most recent event.

40 HP, and every turn can move to any adjacent, passable tile. When moving onto a monster tile, combat is resolved instantly, the monster is removed and the player loses a number of HP. Combat is stochastic: enemies may deal between 5 and 14 points of damage, determined each time the level starts. Moving onto a treasure tile removes the treasure and increases the treasure counter by one, while moving onto a potion tile removes the potion and increases the player's HP by 10 (up to a maximum of 40). If the player moves onto the dungeon exit, the level is completed.

The number of tile types and allowed player actions is very limited, and monsters do not move. Hidden information is only a factor in the game for combat actions, as enemies deal a variable amount of damage, but the damage range is quickly induced after a few rounds of combat. For all purposes, the complete game rules are quickly learned by human players and are simple enough to potentially allow a number of agent construction approaches.

The relatively small size of the level and the fact that it is a discretized space, results in a high decision density. Even a single action, such as moving to an adjoining empty tile, significantly changes the game state in terms of remaining steps to the exit, monsters, potions, and treasures. This means that any input that significantly changes the game state entails a specific decision. The bounded number of affordances in this testbed limit the number of utility sources that must be considered when constructing an agent-persona. Finally,

the small level size means that most playthroughs can be completed relatively quickly.

3.2 Generative Agents

To produce an agent representative of archetypical players, any technique capable of incorporating the concept of a utility function would technically be a possibility. Any reinforcement learning technique satisfies these requirements, including any form of dynamic programming, Monte Carlo methods, and temporal-difference learning [10]. Among them, one-step Q-learing was selected for its simplicity as well as its ability to handle the stochastic nature of combat implemented in the testbed game. Additionally, the small gameworld and limited number of hero moves in each level position permit the use of a lookup table for storing state-action pairs. In an attempt to maintain the Markov property of each state, states in the lookup table consist of the entire gameworld (including passable and impassable tiles, the hero's location and the location of undefeated monsters and uncollected treasures and potions) as well as an abstraction of the hero's hitpoints. The latter is encoded as an integer with 4 possible values, with 0 for 1-5 HP (can certainly not defeat any monster), 1 for 6-14 HP (is likely to die from a monster), 2 for 15-30 HP (can defeat at least one monster) and 3 for 31-40 HP (will not benefit to the full extent from a potion). The addition of these hitpoint ranges to the state description implicitly includes a model of the environment since the enumerators were selected based on the damage range of monsters and the HP healed by potions; although one of the advantages of temporal-difference learning is its ability to operate without a model of the environment, the addition of hitpoint enumerators aimed to speed up convergence of the Q-learning process.

In Q-learning [13], the agent in a particular state s performs an action a (move up, down, left or right) and observes the subsequent state s'. The $Q(s,a)$ value is then increased by $\alpha[r + \gamma max_a Q(s',a) - Q(s,a)]$, where r is the reward in state s', α is the learning rate and γ is the discount factor of future rewards. For training the agents in the presented experiment on a specific game level, $2.5{\cdot}10^5$ games were played with $\alpha = 0.5$ and $\gamma = 0.9$. During training, the action with the highest Q value was selected with a likelihood of $1-\epsilon$ (ϵ-greedy); in the experiments detailed in this paper, ϵ starts at 1 and starts decreasing linearly after 2500 games from $\epsilon = 1$ to $\epsilon = 0.1$ at the end of the training session. When not selecting the highest Q value or in unvisited states, exploration favors the least often taken action in that state.

The reward function of the Q-learning agent is simply the model of the player's utility function. In order to produce multiple different personas for comparisons with players, a number of distinct agents were developed which had different playing styles (see Table 1). All possible outcomes of an action are assigned rewards and each agent (except Baseline) receives a single additional reward; this is expected to create distinct behaviors each emphasizing a particular affordance as a source of utility. While more elaborate strategies with multiple rewards could be included, this paper focuses on "archetypical" agents which are straightforward to understand or modify by designers.

4. DATA COLLECTION

In order to collect human decisions in the form of play-traces in the game environment, a crowdsourcing experiment was conducted. The experiment placed the game on a public webpage which was advertised via e-mail and social media.

Table 1: Description of agents.

Agent (Abbrv.)	Playing Style
Baseline Player (B)	Reach exit.
Runner (R)	Minimize moves.
Survivalist (S)	Minimize risk.
Monster Killer (M)	Kill all monsters in level.
Treasure Collector (T)	Collect all treasure in level.

Table 2: Rewards r for specific game events.

	Agent				
Event	B	R	S	M	T
Killed monster				1	
Was killed			-1		
Reached exit	0.5	0.5	0.5	0.5	0.5
Collected treasure					1
Moved		-0.01			

The starting screen informed the participants that they would be taking part in an experiment concerning computer games, but not its goals of modeling decision making styles. Upon starting, participants had the option of voluntarily providing their name and e-mail address and were informed that participants who chose to do so would enter a lottery and a competition. One participant would be drawn at random and additionally the participant who "did best" would receive a prize as well. In order to ensure variation in the players' concepts of utility, the notion of what constituted *best* was not explained and left to the player's imagination. This design choice was expected to motivate players to exhibit different play styles, i.e. allocating different priorities to reaching the exit of the level, avoiding damage, killing monsters, or collecting treasures and potions. By the same logic, the decision to participate in the competition and lottery was left to the player, since we assumed that this would be of utility to some players and irrelevant to others.

Following a brief introduction on the mechanics and visuals of the game, participants began play on a "tutorial level", which they were allowed to replay as many times as they wished, followed by 10 "real levels" (see Fig. 2), each of which they could play once (i.e. without replays if the hero died). Between levels, players were presented with a summary screen of their previous level, with information on the hero's final HP, monsters killed, treasures collected, potions drunk, actions taken and percentage of level explored. As with the choice of leaving the notion of best performance unclear in the starting screen, showing as many diverse statistics as possible was expected to elicit different play styles among participants. All player actions on every level were logged and stored in an online database.

Apart from the hand-crafted tutorial level, the levels used in the protocol were created via a mixed-initiative design process. Dungeons were generated via constrained genetic algorithms according to the process described in [6], followed by manual adjustments in order to increase the range of interesting, risky actions and the rewards they offer. Most levels have multiple paths to the exit, each path needing different degrees of combat or no combat at all. All levels also have side passages and diversions, with treasures and

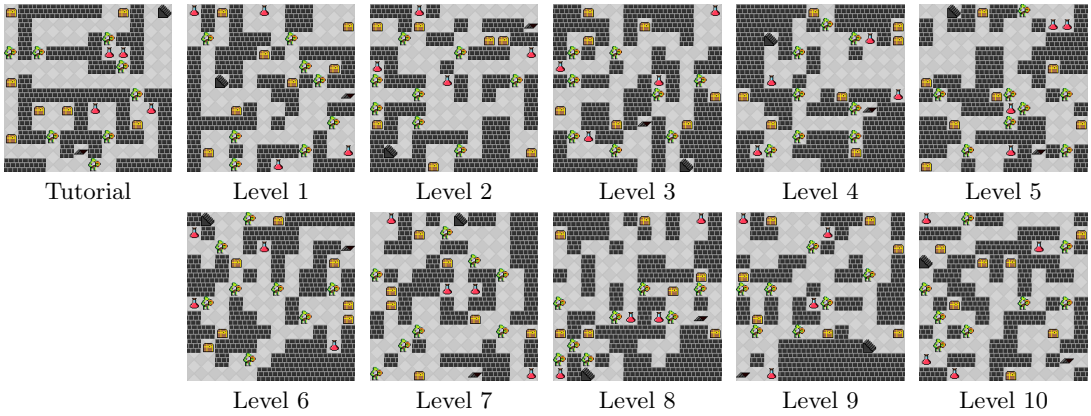


Figure 2: The levels used for the data collections experiment. The tutorial level is hand-crafted, and could be played multiple times. The "real" levels (1-10) were played only once (no retries if the hero died) and were created in a mixed-initiative fashion.

potions often guarded by monsters, but at times also unguarded, either at the end of a long side passage or along a path to the exit. Finally, monsters are usually placed in corridors allowing no way through except via combat; some levels (such as level 8) also include unavoidable monsters on the path from entrance to exit.

4.1 Human Playtraces

38 players successfully completed all 10 levels of the experiment. Some of the most consistent behavioral patterns across players was that of treasure and potion collecting, since both were collected quite consistently by most users.

While treasures were never explicitly deemed important and serve no in-game purpose, the name itself and its significance in many role-playing games plausibly made several players strive towards collecting all of them; the fact that, apart from hit points, treasures collected was the only other statistic visible on the user interface may have also contributed to this. Although not all players targeted treasures, 32 of the 38 players finished the levels with more than 60 total treasures (out of 70). Potions, on the other hand, were often collected by necessity in order to survive combat with monsters which were, for the most part, guarding treasures. As such, it is not surprising that most players collected potions, although there was not as obvious or consistent a drive to collect potions as there was for collecting treasure; out of 38 players, 22 finished the levels with more than 30 total potions (out of 40), and 11 with more than 35.

In terms of actions taken and tiles explored, little variation between players existed, although the (few) outliers are of interest. Two players finished all 10 levels having visited 349 and 395 tiles in total, respectively, which compared to the average 594.4 explored tiles across players indicates that they were trying to complete each level quickly, possibly due to lack of interest or in order to see the next level.

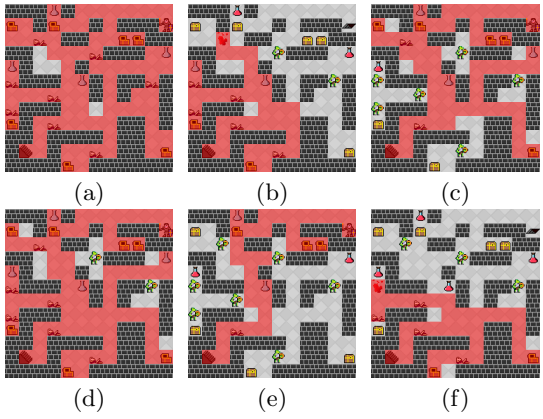
In terms of monsters killed, player behavior was less consistent: since every level contained 8 monsters, even with the help of potions the likelihood of defeating all of them was slim due to the stochastic nature of combat. Due to the fact that each level had different needs for killing monsters (such as unavoidable monsters for reaching the exit), there were few consistent patterns either between players or between levels. The data indicates, however, that players did not explicitly target killing monsters as their goal, possibly because they had no chance of replaying the level if they died. Of the 38 players, only 13 finished the levels with more than 60 total monster kills (out of 80) and only 5 with more than 70. Even players who collected all treasures in all levels did not succeed in killing all the monsters in every level, and no player reached 80 out of 80 monster kills.

An interesting visual aid for qualitatively assessing the behavior of different players is the level's "heatmap", i.e. the tiles visited by the player during her playthrough. Fig. 3 shows some indicative heatmaps of different players on the same level, which illustrate the different player behaviors. Certain players acted as "completionists", and explored most of the level, collected all the treasure, drank all the potions and killed all the monsters (Fig. 3a). Other players rushed to the exit, killing only the minimal number of monsters and ignoring treasures and potions even if they were not guarded by monsters (Fig. 3e). Many players collected the unguarded potions and treasures, and a few guarded ones if the risk was limited (Fig. 3c) while others did not accurately assess the risk involved and died; Fig. 3b and Fig. 3f are particularly good examples of the latter, since the players could have collected the unguarded potions before attacking the monster which killed them.

4.2 Artificial Playtraces

The five generative agents of Table 1 were trained for each level of the user study. Each agent was trained via $2.5{\cdot}10^5$ playthroughs, using the parameters described in Section 3.2. Once training was completed, exploration and learning were disabled ($\alpha = \epsilon = 0$) and 20 test playthroughs of the level were performed to assess the agent's performance — unless otherwise noted, statistics in this section will refer to the average of those 20 playthroughs.

The behavior of the generative agents was largely dependent on the level in which they were trained. Table 3 includes some indicative game statistics of the agents' overall playthrough of levels 1 to 10, which provide some insight on the agents' behavior. In several levels the Baseline (B), the Runner (R) and the Survivalist (S) agents had very similar behaviors as they took the shortest path to the exit (see Fig. 4a, where their heatmap is identical); this was due to the fact that the shortest path to the exit usually did not contain enough monsters to kill the player (which would be detrimental to the Survivalist agent). Despite such similar-



(a) (b) (c)

(d) (e) (f)

Figure 3: Heatmaps of selected players in Level 2. Some acted as "completionists"; others rushed to the exit. Many players only collected guarded items if the risk was limited while others took excessive risks and died. The heatmaps indicate that a single level allows for different decision making styles in spite of the apparent simplicity of the testbed.

Table 3: Game statistics of each artificial agent for the entire playthrough of 10 levels. With the exception of Times Died, values are averaged across 20 test runs; Times Died includes all 200 playthroughs tested.

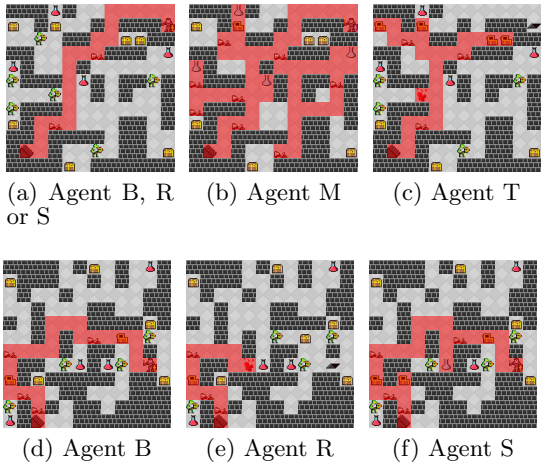
	Agent				
Statistic	B	R	S	M	T
Monsters	22.7	22.6	21.4	53.8	48.2
Treasures	9.4	7.8	11.0	9.4	48.9
Potions	2.1	2.0	3.1	16.1	3.7
Tiles Explored	236	230	244	302	328
Times Died	13	22	0	63	169

ities, agent S did not die in any of the 200 runs (20 runs of each of the 10 levels), while agent B died 13 times, agent R died 23 times, agent M died 63 times and agent T died 169 times. The high death rate of Monster Killer (M) and Treasure Collector (T) agents is due to the fact that, since they were not penalized for dying, the agents took unnecessary risks to kill monsters and collect treasures, respectively. While they were not as thorough in clearing the entire level as human players, agent M finished all 10 levels with 53.8 total monsters killed (out of 80) while agent T finished all 10 levels with 48.9 treasures (out of 70), far more than other agents. Of the remaining statistics it is worth noting that the Runner agent finished all levels with the lowest number of tiles explored, although agents B and S have only somewhat higher values. Finally, the Monster Killer agent collected the largest number of potions in order to survive more combat encounters and achieve more monster kills. The Survivalist agent was also expected to collect a fair number of potions, in order to increase the chance of surviving, but the fact that most levels did not have enough unavoidable monsters between the dungeon entrance and the exit made such a strategy redundant except in special cases (see Fig. 4f).

5. RESULTS

In order to compare player decisions to agent-persona decisions, a simple metric was defined: for each player's playtrace, we replay the whole game and at each point in time, we input the state description to all of our artificial agents, and compare the player's decision to the decision of the different agents. Essentially, we ask: "What would Q do?". This metric expresses the degree of agreement on next best action between the individual player and the agent-persona. It is directly grounded in the theoretical considerations of decision making outlined above, and assumes that for every given state of the game, an agent-persona that is adequately representative of a player in that particular state will select the same action as the player. More precisely, the metric was calculated as the number of agent-persona/human player agreements N_a for each decision made in the human decision trace, normalized with respect to the number of decisions in the player's decision trace N, i.e. N_a/N. One advantage of this metric is that it gives a numeric representation of the degree to which an agent adequately represents a player across a level. The utilities of each agent could subsequently be tweaked through iterations of training using a simple hill-climbing approach to maximize the agreement ratio with regard to an individual player or to clusters of players. Another advantage is that the agreement ratios would be easily intelligible to game designers using the agent-personas in a content creation process. In order to test the agent-personas as well as the comparison metrics, a Random Controller was constructed which chose randomly from all legal moves from each game state. This addition investigates to which degree the agent-personas decided and represented players differently from a random agent.

For each level in the user study, each playtrace was examined to determine which agent-persona had the highest agreement, and hence represented the best fit for the playtrace. Table 4 indicates the number of times each agent was the best fit for each level. As is evident from the table, most playtraces matched the Treasure Collector (T) persona, while subgroups of players matched other personas. This finding corroborates the observation in Section 4.1 of players' tendency to collect treasures, evidenced by the large proportion of players that collected most (and some all) treasures across levels. This behavior may have stemmed from



(a) Agent B, R or S (b) Agent M (c) Agent T

(d) Agent B (e) Agent R (f) Agent S

Figure 4: Some indicative heatmaps of trained agents on Level 2 (Fig. 4a–4c) and Level 8 (Fig. 4d–4f). The different playstyles of the agent-personas are showcased, although in Level 2 agents B, R and S all share the heatmap of Fig. 4a.

Table 4: Frequencies of agent-persona best fit across levels.

	Agent						
Level	B	R	S	M	T	Z	Total
1	1			10	27		38
2	2	4		5	27		38
3		5	2		31		38
4	1	2	1		34		38
5		3			35		38
6		3		4	31		38
7	6	3		7	22		38
8	1		4		33		38
9	3	2		1	31	1	38
10	1	7		2	28		38
Total	15	29	7	29	299	1	380

Table 5: Statistics of the individual agent-personas. All agent-personas attain high maximal values. This indicates that all agents, except for the random controller, are relevant approximations of some players.

Agent	Mean	SD	Max	Min	N
Baseline Player (B)	0.52	0.10	0.94	0.25	15
Runner (R)	0.54	0.09	0.94	0.37	29
Survivalist (S)	0.53	0.11	0.94	0.25	7
Monster Killer (M)	0.54	0.10	0.80	0.23	29
Treasure Collector (T)	0.63	0.11	0.90	0.35	299
Random Controller (Z)	0.43	0.02	0.49	0.37	1

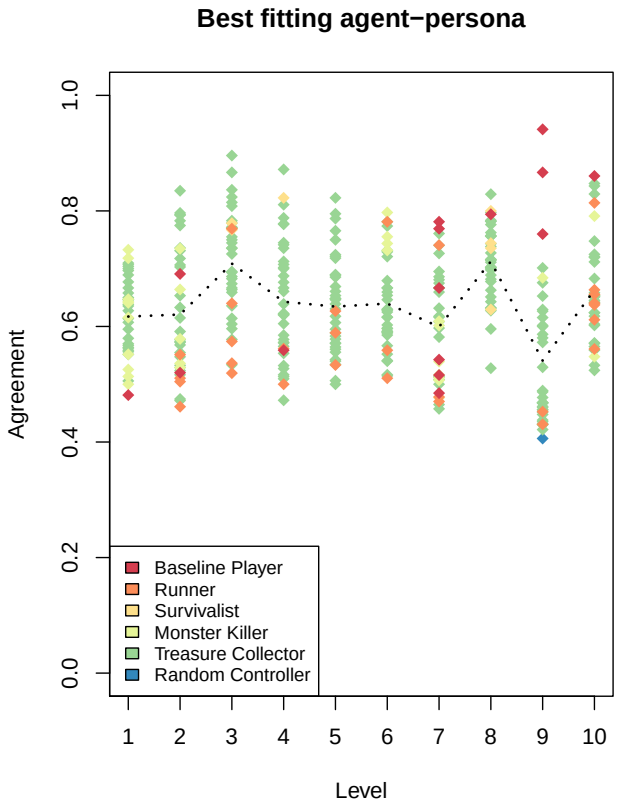


Figure 5: The best agreeing agent for each level. The dotted line indicates the mean agreement of the best fitting agent-personas across all players on each level.

the treasure counter on the user interface as well as the encouragement of being the "best" in the game. Unfortunately, no post-play qualitative data were collected, which could have helped illuminate individual motivations of players. Although the Treasure Collector persona does seem to dominate the dataset in terms of agreements, all other agents except for the Survivalist have a strong minority representation as best fits. The general relevance of the method is supported by the fact that only a single playtrace was characterized best by the Random Controller (Z).

In order to assess the performance of the best fitting agent-personas for each playtrace, the agreements are visualized in the plot depicted in Fig. 5. The plot shows how the agent on average agreed with players on 60%–70% of their decisions. A Mann-Whitney U test unsurprisingly indicated that collectively, the best-fitting agent for each playtrace agreed significantly more with players than the Random Controller ($W=155633.5$, $p<0.001$).

Table 5 summarizes the performance of each agent across all levels. The results show that all agents attain a high level of agreement with some playtraces and very low levels of agreement with others. This indicates a variety in the expressed utility functions of the agent-personas, but the fact that the Treasures Collector agent dominates the data set in terms of best fit suggests that this agent possibly could be split into multiple agents to better represent the playtraces for which it is the best fit.

6. DISCUSSION

The method developed and demonstrated in this paper seems to have a number of attractive characteristics, allowing for the construction of decision making personas and determining to which degree different players agree to them. However, this method also suffers from a number of limitations warranting further work. These limitations concern the data collection method, the agent as an abstract model of human decision makers, and the scalability of the computational approach.

The applied data collection method sought to enable players to engage with the game in accordance with their individual motivations and hence utilities. As part of this goal, players were given the option of participating in a competition and lottery. The collected data exhibits a predominance of behavior matching the Treasure Collector persona which could be a consequence of players trying to win the competition.

The utility function of the agent-persona is constant over the course of each level, and only one agent-persona is used to characterize a full decision making trace. This means that if the player changes her conception of utility while playing the level, she will quite possibly match several personas during the playthrough. A response to this limitation could be to subdivide decision traces, e.g. via a sliding window, to find the best agent-persona match for each point along the decision making trace. As an extension, this approach could be used to cover all playtraces for an individual human player to investigate which personas are matched across all the player's traces. Relatedly, the utility function of the agent will necessarily be a high-level abstraction of the player's. While this is intentional, other factors influ-

encing the agent's evaluations and learning, such as exploration chance, learning rate and γ value (discount of future rewards) are kept constant in the experiment presented here. Each of these could have, at some level, relevant psychological counterparts such as openness to new experience, ability at learning rules and content, and tendency and motivation to plan ahead; the extend to which these parameters map to human psychology should be explored, and agents with different configurations of these parameters should be tested.

The testbed game used for the development and demonstration of the method has a limited number of affordances that are considered potential components of the player's utility function. Hence, the construction of various agent-personas based on various configurations of these is a manageable task, which can be done manually. For more complex games, the number of affordances may be significantly higher, making it difficult and time consuming to construct agents that cover the space of possible utility configurations to a degree that a good agent-persona match could be found for every player. This affects the scalability of the method, albeit the degree remains unknown at this point. One possible solution could be to use the method for modeling players at a conceptual level and designing content at a sketch level, rather than at a detailed level, though this will naturally depend on the game in question. While the Q-learning agents were demonstrated to work well, the training of the agents is computationally demanding and hence time consuming. The time needed to train the Q-learning agents on an average desktop computer would likely exceed the time a content designer would be willing to wait for agent-based feedback. A better approach would be to use a generic trained agent, whose policy was not tied to a particular level. Possible approaches could include using agents based on Q-learning with neural networks, Monte Carlo Tree Search or evolutionary rule-based systems.

Future work will focus on addressing these limitations, in an attempt to find a faster performing, more accurately representative, and scalable approach to modeling human decision making in the form of generative agents. We will also attempt to adapt the a priori constructed agents to fit either individual players or, more realistically, generalized representations of players. Such generalized representations could be obtained by clustering players based on their difference from the various agent-personas, and training the closest agent-persona to match the center of the cluster [5].

7. CONCLUSION

This paper presented a theory-based method of using generative agents as models of human decision making in computer games and explored it in a simple scenario. A theoretical argument for considering agents eligible for representing variations in human decision making processes as agent-personas was presented. To test this argument, a crowd-sourced human decision making experiment was conducted using a testbed game. A number of Q-learning agents were developed as agent-personas, and the decision making of human players was compared to the decision making of the agents. The comparison demonstrated that the agents were useful as personas for characterizing and discriminating between the human players. Although the suggested method has a number of limitations in its current form, key findings demonstrate that a high-level abstraction of human decision making, in the form of agents, is possible and can provide useful insights on possible and plausible interactions with game levels, whether hand crafted or procedurally generated. We believe that the method could be of use to player modeling as well as game design and development.

8. ACKNOWLEDGEMENTS

We thank the participants of the user study. The research is supported, in part, by the FP7 ICT project C2Learn (project no: 318480).

9. REFERENCES

[1] D. Borsboom. *Measuring the Mind: Conceptual Issues in Contemporary Psychometrics*. Cambridge University Press, 2005.

[2] D. Borsboom, G. J. Mellenbergh, and J. van Heerden. The Concept of Validity. *Psychological Review*, 111(4):1061, 2004.

[3] A. Canossa and A. Drachen. Patterns of Play: Play-Personas in User-Centred Game Development. In *Breaking New Ground: Innovation in Games, Play, Practice and Theory: Proceedings of the 2009 DiGRA Conference*, 2009.

[4] J. Gibson. The Concept of Affordances. *Perceiving, Acting, and Knowing*, pages 67–82, 1977.

[5] C. Holmgård, J. Togelius, and G. N. Yannakakis. Decision Making Styles as Deviation from Rational Action. A Super Mario Case Study. In *Ninth Annual AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 2013.

[6] A. Liapis, G. N. Yannakakis, and J. Togelius. Towards a Generic Method of Evaluating Game Levels. In *Ninth Annual AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 2013.

[7] C. H. Pedersen and G. N. Yannakakis. Epistemological Challenges for Operationalism in Player Experience Research. In *Proceedings of the 7th International Conference of Foundations of Digital Games*, 2012.

[8] A. Rubinstein. *Modeling Bounded Rationality*, volume 1. MIT Press, 1998.

[9] A. M. Smith, C. Lewis, K. Hullett, G. Smith, and A. Sullivan. An Inclusive Taxonomy of Player Modeling. *University of California, Santa Cruz, Tech. Rep. UCSC-SOE-11-13*, 2011.

[10] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. Cambridge University Press, 1998.

[11] A. Tversky and D. Kahneman. Judgment under Uncertainty: Heuristics and Biases. *Science*, 185(4157):1124–1131, 1974.

[12] A. Tychsen and A. Canossa. Defining Personas in Games Using Metrics. In *Proceedings of the 2008 Conference on Future Play: Research, Play, Share*, pages 73–80. ACM, 2008.

[13] C. J. C. H. Watkins and P. Dayan. Q-learning. *Machine Learning*, 8(3-4):279–292, 1992.

[14] G. N. Yannakakis, P. Spronck, D. Loiacono, and E. André. Player Modeling. In *Artificial and Computational Intelligence in Games*, pages 45–55. Dagstuhl Publishing, Saarbrücken/Wadern, 2013.