

Integrating Formal Qualitative Analysis Techniques within a Procedural Narrative Generation System

Ben Kybartas and Clark Verbrugge
School of Computer Science, McGill University
ben.kybartas@mail.mcgill.ca
clump@cs.mcgill.ca

Abstract

Qualitative analysis of procedurally generated narratives remains a difficult hurdle for most narrative generation tools. Typical analysis involves the use of human studies, rating the quality of the generated narratives against a given set of criteria, a costly and time consuming process. In this paper we integrate a set of features within the REGEN system which aim to ensure narrative correctness and quality. Correct generation is ensured by performing an analysis of the preconditions and postconditions of each narrative event. Narrative quality is ensured by using an existing set of formal metrics which relate quality to the structure of the narrative to guide narrative generation. This quantitative approach provides an objective means of guaranteeing quality within narrative generation.

Introduction

Procedural Narrative Generation refers to process of dynamically creating stories without, or reducing, the need for a human author. The ideal system would aim to provide an infinite set of interesting and diverse narratives at no cost to the user of the system. The development of such systems, however, tend to be complex and costly, and can often lead to narratives which are simplistic or of low quality.

For a narrative generation tool to be successful and usable within a game, there are generally two features which must be present. First, the narrative must be valid. This means that the narrative must be a complete sequence of actions which could feasibly occur within the world in which the narrative is set. The second property, and the more abstract and therefore complex, is that the stories are preferably of high quality. While most narrative generation tools satisfy the requirement of formality, the number of generation tools which contain embedded quality evaluations have been minimal. Quality of narratives is typically determined by means of a human-study of a collection of generated narratives. However, this is a costly step and would be infeasible for commercial developers wishing to use the system.

In this paper, we present several methods for integrating validation and quality within the narrative generation tool itself. We build atop the REGEN system, originally presented

in (Kybartas 2013). The REGEN system is a narrative generation tool which utilizes a graph rewriting process to generating narratives. It does so by incrementally applying a series of rewrite rules to a graph representation of the game narrative, while basing the rewrite rules off of subgraph patterns within a graph-representation of the game world. Our focus is on modifying the existing system in order to add a formal validation step and additionally utilize a sequence of graph-analysis metrics to gauge the quality of the narrative structure and guide it towards optimizing these metrics. As an addition we present a more abstract analysis of narrative quality by examining how the game world structure is able to influence the narratives generated by the REGEN system. Although by no means a replacement for human study, our aim is to provide a basic tool to help users analyze structural quality automatically, assisting in the creation of better narratives. The contributions of this paper include:

- Based on a set of metrics measuring structural narrative quality, we develop a guided generation process that attempts to optimize for quality by guaranteeing metric thresholds throughout generation.
- Our generated narratives are incrementally validated for correctness, ensuring all narratives can viably exist within the current context of the game.
- We explore how the structure and scale of the game world (state) influences the quality measures in the generated narratives.

Below we first present related work in the field of procedural narrative generation, including a summary of the design and functionality of the existing REGEN system and the quality metrics it defines. We then present how the system is modified to use validation and metric-enhanced generation. In our experimentation, we show the effects of each modification to the system, and further present an experiment wherein we analyse the effects of the game world construction on the quality of the narratives generated.

Related Work

Approaches to qualitatively analysing narrative generation systems are often varied. A human survey remains a popular means of allowing reviewers to analyse a set of generated narratives according to a given set of metrics defined by the creators of the system. Often these experiments

have the participants blindly compare generated narratives to manually written narratives. Barber and Kudenko utilized their GADIN system as a soap-opera generator, and evaluated it using a Turing test where participants were asked to choose which narrative they felt was generated, and which was hand-authored (Barber and Kudenko 2007). In Peinado and Gervás' experiment, the generated narrative was compared against a hand-authored narrative and a narrative composed of random story events. The participants were asked to rank the narrative according to *linguistic quality*, *coherence*, *interest* and *originality* (Peinado and Gervás 2006). Pérez y Pérez *et al.* attempt to analyze *novelty*, defined similarly to the *originality* metric, between the generated narratives to ensure each new narrative was different from the one that preceded it (Pérez y Pérez *et al.* 2011). Verbrugge and Zhang modelled narratives as Petri-nets, and used this to evaluate various properties of narrative structure, similar to our analysis of the graph structure of narratives (Verbrugge and Zhang 2010).

For our system, we utilized metrics that define quality based on the structure of the narrative, and embed this knowledge within the system. Certain metrics we use are similar to the ones previously used for human surveys, for example we use a *uniqueness* metric similar in definition to the previously mentioned novelty and originality metric. The IPOCL system and Porteous *et al.*'s generation tool both constrain their generation process to achieve *authorial goals*, which are specific story events or outcomes which are specified by the author (Porteous, Cavazza, and Charles 2010; Riedl and Young 2010). In this sense, the quality of the story hinges upon these conditions being met, and the authorial conditions being met result in the author's vision of a high quality story. The IPOCL system, as with REGEN and many of the "emergent narrative" generation tools also structure their systems in such a way that each action taken by the characters within the story feels logical (Cavazza *et al.* 2009; Chang and Soo 2009; McCoy *et al.* 2010; Mateas and Stern 2005). This often done by modelling the character's personality and/or relations to other characters within the world and letting these directly influence the progression of the narrative. Within these systems, having the narrative be believable is directly related to generating a narrative of high quality. While maintaining this functionality, our aim in this paper is to integrate the quality metrics typically analyzed by human surveys directly into the generation system.

Other qualitative analyses have been focused around particular aspects of narratives, often with relations to story conflict and tension. Ware *et al.*, for example, describe four formal metrics aimed to measure the quality of conflict within a narrative, focusing on issues of balance, directness, intensity and resolution (Ware *et al.* 2012). CPOCL embedded a formal model of conflict within the narrative planner, implying that gearing narrative generation tools towards conflict would create more realistic and engaging narratives (Ware and Young 2012). Using genetic algorithms, Giannatos *et al.* attempted to generate narratives that optimize the story's suspense, where the quality of the narrative is a direct result of the level of suspense (Giannatos *et al.* 2012).

Our metric analysis provides some preliminary and basic indications of conflict within the narrative, although these are basic and the intention is instead that conflict emerges naturally from the interaction between the game's characters.

The REGEN System

The REGEN system uses graph rewriting techniques for the main generation process. Graph rewriting is a generalization of the string rewriting strategy commonly associated with computer language grammars. In graph rewriting one defines rules that search for patterns in graphs (as opposed to strings), rewriting the resulting matched areas to produce a new graph (Ehrig *et al.* 1999). Within the system, this involves representing the *game narrative* formally as a graph and then applying rewrite rules to this graph in order to generate a more structurally complex narrative. The core purpose of the system is to allow the narratives to make changes to the *game world*, and have all subsequent narratives be respectful of these changes, giving the player the sense that their actions have meaningful purpose within the game world. Likewise, these effects can lead to specific narratives where certain events only occur because of the actions the players have taken in previous narratives. Note that we only present a basic description of the system within this paper, for a more formal and detailed description of the functionality of REGEN please consult the thesis work (Kybartas 2013).

In REGEN, the *game narrative* is represented as a directed acyclic graph where each node represents a narrative event and the edges indicate the ordering of the events. Each narrative has a starting node, and an ending node. The narrative experienced by the player refers to the path taken by the player from the start node to the end node, where completing the event of one node allows the player to travel to any of the nodes linked to by that node. This means that in any instance where a specific event leads to two or more events, then we have a "branching" narrative, a common feature in video game narratives where the player is given different possible paths through the narrative where at specific events, they may make the choice of which action to take next.

The *game world* is represented as a directed graph with labelled edges. In this instance, each node represents an object in the game world, such as a non-player character (NPC), object or location. The edges between objects show the relation from one object to another. For example, an NPC may hate another NPC, or own a particular object, etc. The game world is important in that we generate narratives by looking for patterns within this game world, such as generating a quest wherein the player murders an NPC who is hated by another NPC at the request of this second NPC.

This main generation process involves two main steps. The first step involves generating the starting narrative with a few general actions, enough to constitute a basic narrative. It is important that this narrative have a defined beginning and end and be completable. The first step of generation is done using a set of rules which are called the *Initial Rewrite Rules* (IRR). The IRR checks for patterns in the game world that could lead to specific narratives, such as the aforementioned

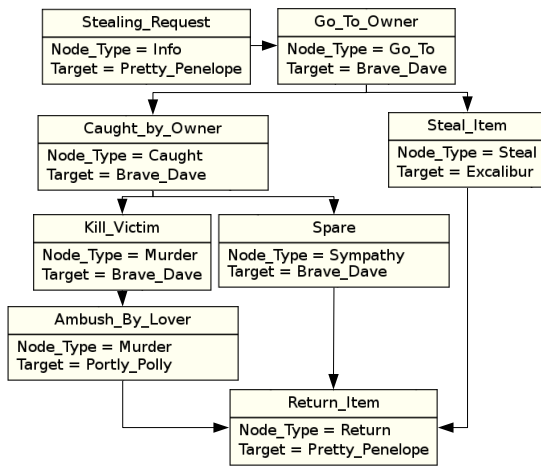


Figure 1: A narrative which started as simply stealing an item. Rewrites allowed for the player to be caught while stealing and have to murder or spare the owner of the item. Killing the owner forces the player to also kill their lover.

hates relation. If these relations exist then the resulting narrative is generated. The pattern and resulting narrative are both defined by the author using the system.

The following step uses a second set of rules called the *Secondary Rewrite Rules* (SRR). In this phase we once again look for patterns within the game world, but we further look for patterns that could result in changes to specific narrative events. For example, a loved one may come to defend the murder victim forcing the player to kill both the lover and the victim. For the SRR the author defines a narrative event that can be rewritten, as well as the social pattern which allows this rewrite and the resulting narrative event(s) that will replace the narrative event being rewritten. After applying a set number of SRRs the narrative is complete, with the number of rewrites being chosen by the user using the system. An example of a “Steal an Item” narrative after being rewritten using secondary rewrite rules is shown in Figure 1.

Qualitative Analysis Techniques

In the following section we present both our formal analysis techniques, and how they are used within the REGEN system. We first present the “metric enhanced rewriting” technique, which uses a set of metrics to guide the generation process towards creating narratives of a certain quality. Secondly, we present a narrative validation approach, which analyzes the pre and post-conditions of the narrative graph to ensure story completeness and validity. We define a *complete* narrative as a narrative wherein all possible paths through the narrative are available to the user. A *valid* narrative means that the narrative may logically exist within the game context, without violating any of the constraints within the game world.

Metric Enhanced Rewriting

In the original version of REGEN, valid SRRs were selected at random during the second stage of generation. To guide

this step towards generating higher quality narratives we present a more sophisticated, “metric enhanced rewriting” rule selection process. Enabling metric enhanced rewrites allows the user to specify a set of metrics and associated weights, with the system then prioritizing rewrites that maximize the weighted result.

Metrics While arbitrary metrics can be defined and used, our current implementation of metric enhanced rewriting relies on a suite of narrative metrics built into REGEN, previously shown to be related to narrative quality. A more complete description of the metrics may be found in (Kybartas 2013); note that each metric maps a narrative to a positive, real value.

Longest Path and *Shortest Path* refer respectively to the longest sequence of narrative events that will take the player from the starting node to the ending node and the shortest sequence of narrative events that reaches the ending node.

Highest Cost and *Lowest Cost* are metrics used to measure the cost of the narrative in terms of consuming non-replaceable resources in the game world. For example, by murdering an NPC or destroying an item, this object is removed from the game world and will no longer be able to be used in later narratives. We define this as a cost action. While having more cost actions gives the player a greater effect on the game world, it does reduce the life-span of the generation tool, as after a certain point we could end up in a game world in which no valid stories can be created.

Number of Branches refers to the number of *branches* in a narrative. We define a branch formally as being any event in the narrative graph which is linked to two or more events counts as a branch, with the player being able to choose the next event.

Encounters were defined as an alternative to cost actions. An encounter is an event which does not remove resources from the game world, but still involves similar actions to a cost event. For example, many role-playing games have an infinite supply of nondescript enemies, generated as needed. Fighting an enemy may be similar to murdering an NPC, but because there are infinite enemies we label it an “encounter” event instead of a cost event.

Uniqueness refers to the number of events within a narrative which are (relatively) “unique”. For example, a quest to kill one enemy would achieve a uniqueness of 1, but a quest to kill one hundred enemies would have a uniqueness of only 0.01 since only 1 out of the hundred murder events would be unique. Uniqueness is important to narrative quality since repetitive actions are often viewed as a detriment as players eventually get tired of performing the same action repeatedly with no variation.

Two additional metrics are available in REGEN, *narrative richness* and *weight of choices*, but could not be optimized for within the rewrites of a single narrative, as they measure inter-narrative properties. We describe them here as they give a sense of how our approach also affects the overall, multi-narrative experience a player may undergo.

Narrative richness gives the percent of narratives which occur due to player action within previous narratives. This is performed by comparing the accumulated preconditions

for a given quest to all the postconditions of the narratives which preceded it. If any of the preconditions are generated by the postconditions of the previous quests, then we have a narrative richness of one, and zero otherwise.

Weight of choices is a metric specifically used in simulation. For simulation we create different game worlds for each possible path the player can take within a narrative and then continue the generation process on each world (to avoid making assumptions about player behaviour). At the end of our simulation, we take the inverse of the average similarity between game worlds to get the proportion of the game world which can be affected by due to the choices made by the player in each narrative.

Weighting and Selection Process The full process of metric enhanced rewriting requires the user specify some (sub)set of metrics, along with associated weights. Weights are simply an integer value which is used to indicate the importance of each metric, with a positive integer indicating that a higher value in that metric correlates to improved narrative quality, while a negative number indicates lesser quality.

Metrics and their weightings are then used at every rewrite step in order to select a rewrite rule. The system first gathers all the possible narratives based on every valid SRR which may be applied; these narratives represent what our main narrative would look like if that particular rule is applied. Each of these possible narratives are then evaluated according to the metrics requested by the user, resulting in a (weighted) score for each metric and potential narrative combination, normalized with respect to the best possible value for that metric. Let n be a narrative from our set of potential narratives N , m a metric mapping narratives to real values, and $w(m)$ the weighting of m . Let $\hat{n} \in N$ be such that $m(\hat{n}) \geq m(n)$, $\forall n \in N$. Equation 1 shows the scoring function:

$$score(n, m) = \frac{m(n)}{m(\hat{n})} \times w(m) \quad (1)$$

This step is repeated for each of the metrics in question. The final score for each narrative is then just the summation of all the individual scores for each metric over the set of metrics M .

The last step is simply to take the narrative with the highest final score as being the “best” narrative. The process repeats for each iteration of the rewrite process, greedily selecting the best narrative at each rewrite stage. While this solution does not guarantee the ideal narrative, aiming instead to achieve a local maximum for the metrics provided, it is a fast and simple method that still allows the user to tweak the generation process towards creating a narrative of a certain quality.

Narrative Validation

Although the rewriting process guarantees a properly formed narrative structure, it cannot guarantee that the resulting narrative is feasible—certain paths through the narrative may imply mutually exclusive world conditions which even if locally valid cannot be combined in a single quest. We thus include a narrative validation phase after each

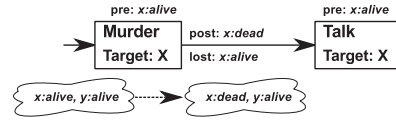


Figure 2: A narrative that would fail validation.

rewrite stage, implemented by first associating game world conditions with events, and then propagating these conditions through the narrative to verify correctness. If this validation fails, we reject the narrative and attempt to validate the next best narrative. If all validations fail, then we finish the rewrite process and return the story as is.

Conditions Each narrative event has necessary conditions on the game world that it requires to execute properly, as well as conditions it ensures or disallows following completion. A murder event, for example, requires the condition that the victim is alive prior to the event, guarantees the condition that the victim is dead after, and also results in the removal of all conditions that may constitute relations between the deceased and other characters in the game world. In REGEN, properties such as these are defined based on a fixed vocabulary, giving us a well-defined set of primitive conditions we can associate with each narrative event as *pre-conditions*, *post-conditions*, and removed or *lost conditions*.

Narrative analysis is then a process of verifying that the potential game world state is acceptable to each narrative event. This is done by propagating conditions through the narrative flow. We accommodate story branching in this process by considering each possible path through a narrative separately.

Propagation and Validation Correctness of the narrative is ensured if the pre-conditions of each event must be true at the start of the event. This, however, depends on the accumulated game state that can reach an event. A forward pass thus propagates post-conditions, filtering out lost conditions, and checking that pre-conditions can always be guaranteed. Let e be a narrative event other than the first one, and \check{e} its predecessor. We then compute

$$out(e) = post(e) \cup (out(\check{e}) - lost(e)),$$

and verify that

$$pre(e) \subseteq out(\check{e}).$$

For the first event in our narrative, we assume $out(\check{e})$ is the current game world state.

Figure 2 gives a simple example of this process. In this case a murder of NPC X precedes a talk event for the same NPC. Propagation of the game world state (bottom) through the first event results in an *out*-set which is not a superset of the pre-conditions of the next event, indicating a problem.

Note that this is only intended to guarantee internal consistency and rationality of the narrative generation process. The player may still manage to cause narrative failure during gameplay by performing actions external to the narrative flow (such as killing a character who is needed later in the narrative). Detecting and handling quest failure is thus still a necessary feature of the game engine.

Experimental Analysis

In this section we present two experiments. The first experiment explores the effect of game world construction on the narrative quality without using metric enhanced rewriting or narrative validation. Following this, we explore the effect of integrating metric enhanced rewriting and narrative validation.

Impact of Game World

As a rewrite system, the output of REGEN clearly depends on the number and kinds of rewrite rules defined. Less well understood, however, is how the game world graph influences quality. Here we thus explore differences in narrative quality for three different types of game world graphs. The first two graphs were hand-authored and the third graph was based of the game world of SKYRIM, adapted into the REGEN system.

Game Worlds The first hand-authored graph consists of a small world graph with *sparse* relations, intending to test our system with minimal data. In total there are 25 entities (NPCs and objects) and 61 edges, averaging around 2.5 edges per node.

The second hand-authored world graph was larger with a *dense* set of relations, intending to test our system with a larger body of data. This graph was designed with 63 entities and 928 relations, averaging just under fifteen edges per node. This gives a heavily connected graph in which most NPCs were designed to have at least one relation to every other NPC object in the game world. We expect narratives of a higher quality for the dense as opposed to sparse world, since with more relations there should be more valid narrative rewrites available to the system.

Lastly, the SKYRIM game world was intended to test our system with a game world at the approximate scale of a commercial game. The SKYRIM world was a conversion of SKYRIM data into our game world format. In order to provide an accurate representation of the SKYRIM world, all data was taken from the *Skyrim Creation Kit*, which provided direct access to the existing SKYRIM relation graph. The relation graph and game world are constructed in a similar fashion to our own game world, modelling relations between NPCs, and was easily converted into our format. We did not include any of the procedurally generated enemies within the SKYRIM world as random encounters are typically kept out of the game's quests and occur instead as a natural consequence of exploring the game world. In the end this amounted to a world graph containing 757 nodes and 1360 edges. Of interest is that this amounts to only 1.8 edges per node, meaning that the density of this graph is similar to that of the Sparse World graph we defined above.

Criteria Interpretation of the metric results on these game worlds requires some criteria for defining quality in terms of metric values. The criteria we have selected are based on our own notions of quality, as the human evaluation of our metrics remains future work at this time. We value larger path length as being positive since it increases the duration of the quest for the player. A higher number of branches is also positive since allowing for player choice adds an element of

player control, reducing the sensation of playing through a static experience. Cost is a negative metric, since high cost quests reduce the lifespan of the system, while encounters are positive for the opposite reason. A high uniqueness is a positive feature since it indicates that most actions within the narrative are unique. Higher narrative richness is good since it implies that the narratives being generated are largely influenced by the actions taken by the player. Lastly a high result for weight of choices is again positive, since it indicates that player actions can significantly affect the game world.

For comparing the results of the three different game worlds, we generated one hundred narratives for each world and then evaluated them using our metrics. We used ten initial rewrite rules and eight secondary rewrite rules, all predefined in REGEN. Examples of the IRRs include murdering players, fighting monsters, and stealing items. Examples of the SRRs include sparing murder victims, being ambushed by loved ones and getting caught while stealing.

Results Figure 3 shows the results for each of the metrics for each of the three game worlds. For the first 5 (intra-narrative) metrics, the dense world outperformed the sparse world given our criteria, although in most cases the differences are not large. Since the two worlds differ significantly in their construction and size, but a similar distribution of relations exists in both, this suggests that the way the rule set works with the distribution of relations is more important than the density of relations.

The SKYRIM game world graph gives dramatically lower results in nearly all metric scores. Examining the structure of the SKYRIM game world graph gives some insight into these generally lower metric scores. First, the density of relations was even less than the sparse world, being on average only 1.8 relations per entity. Secondly, the rule set we defined focused heavily on working with the extreme relations, such as *hates* or *loves* relations. However in the SKYRIM game world graph there are only 41 loves and eight hates relations out of the 1360 total relations.

Much more common were milder relations, such as *friends* relations, which could be attributed to the fact that the SKYRIM game world was aimed at defining AI behaviours, as opposed to it being intended for narrative generation purposes. Overall, this indicates that when using rewrite systems for narrative generation it is important to either tailor the rules to work with the predominant relations, or tailor the game world to closely match the rule set. It is reassuring, however, that we are still able to get decent results using the SKYRIM game world graph without any changes to our existing rule set and we believe that tailoring the rules to more closely reflect the relation distribution of the SKYRIM world would improve these results.

Interestingly, both narrative richness and weight of choices were highly in favour of the sparse world. This is likely because the small size allows for intense player interaction and consequence. With a sparse set of relations, over time we expect that many of the relations will end up being relations created due to player actions, leading to higher values for both metrics, and this is reflected in the results. The effects are also seen in weight of choices: the player

Metric Ranges for Different Game Worlds

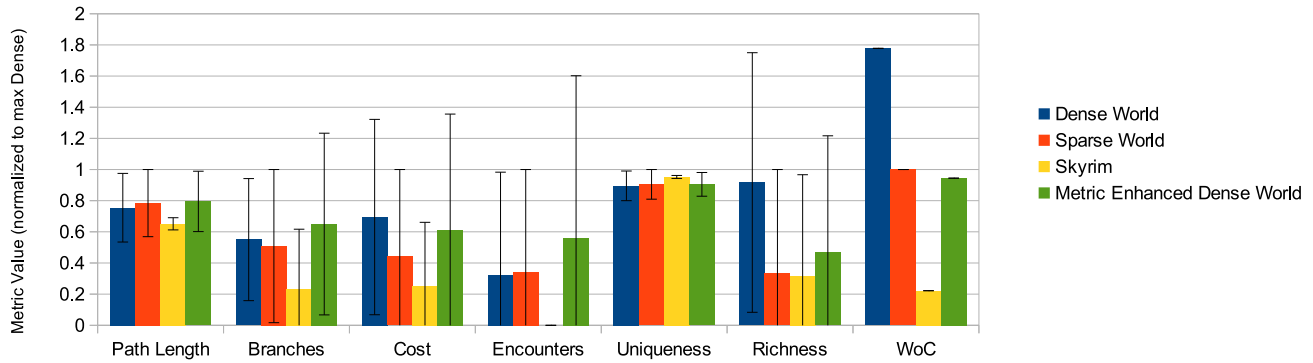


Figure 3: Metrics results on narratives. Each bar shows the average of the metric values measured, with the error bars indicating the standard deviation. Each metric result is normalized to the maximum value of the dense world (average + standard deviation). Note that *weight of choices* (WoC) has no standard deviation.

can generally affect more of the sparse game world with the choices they make, when compared to the dense and SKYRIM worlds. A small world with many narratives necessarily encourages more visible impact, and so these scores could be improved for the larger game worlds by generating more narratives. Note, though, that one hundred is a fairly large number of quests for a player to complete in any current single-player game, and to achieve similar metric results in SKYRIM-scale game world would require many thousands of quests.

Lastly, it was noted that the standard deviation was quite large on most of the metrics, indicating a large variation in narrative quality. While this is an indication that low quality narratives are occasionally being generated, it is important to note that certain narratives may be high in one metric, but have little or no score in another metric. Future work can aim to reduce this variance, most likely with the addition of more rewrite rules.

Metric Enhanced Rewriting and Narrative Validation

Figure 3 also gives results of a metric enhanced rewrite process applied to the dense world, shown as the last bar in each set. These results were generated using weights of 1 for uniqueness, 2 for encounters, 4 for path length, 1 for cost and 5 for number of branches, values intended to approximate the relative emphases of our quality criteria. This resulted in a noticeable improvement in our metric results, at least for the specific, intra-narrative metrics these weights affected.

Inter-narrative metric results imply an increase in quality with regards to the richness but a decline in quality with regards to weight of choices. The decline in weight of choices could be related to the increase in encounters, which have only a very minor effect on the game world, although this decline is only minor.

Interdependencies also exist within the weighted metrics themselves. In experimentation we found that using negative values for cost would reduce the number of branches due to the fact that many of the branch rewrite rules introduced cost actions. Likewise, we ended up assigning uniqueness a lower weight, since while doing so allowed for more repeated actions, it also resulted in longer narratives. The resulting trade-off balances our quality criteria—quality of the narrative slightly declining in terms of cost, uniqueness and weight of choices, but with much more significant improvements relating to path length, branches, richness and encounters.

In terms of validation, throughout experimentation we monitored the number of successful rewrites versus unsuccessful rewrites. The percentage of invalid rewrites tended to fluctuate between 0 and 15%. These results stress the importance of the validation step, since otherwise there would be instances when the tool would generate incorrect narratives up to fifteen percent of the time. By performing this step, we can catch these narratives and reject them during the generation process, rather than leaving them for the player to discover impossible during gameplay.

Conclusion

In this paper, we provided several techniques to guide the REGEN system to create narratives geared towards a specific quality. These qualities were aimed at both generalized metrics that are often only evaluated through human surveys, and also fundamental validations that ensure consistency between the narrative and the game world. Here we integrated the former into a more sophisticated rule selection mechanic, and guaranteed the latter through a formal validation approach also embedded into the generation process. This results in improved narratives, guaranteed to be valid in the current game world.

We also examined how two of the more abstract met-

rics, relating to how the game world and generation process shapes over time, relate to the construction and size of the game world. The results for this experiment show that narrative richness and weight of choices are improved by having small game worlds. This, however, will directly conflict with the size of commercial scale game worlds, such as the SKYRIM world used. Further research in this area would be to integrate an improved generation system which also aims to optimize these metrics, guiding the generation tool to achieve similar quality results regardless of the size of the game world. Additionally, a human study comparing several narrative quality assessments against our automatic assessments would further strengthen our argument that the metrics are accurate representations of structural narrative quality and this remains future work for the system.

References

- Barber, H., and Kudenko, D. 2007. Generation of dilemma-based interactive narratives with a changeable story goal. In *Proceedings of the 2nd international conference on INtelligent TEchnologies for interactive enterTAINment*, INTETAIN '08, 6:1–6:10. ICST, Brussels, Belgium, Belgium: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).
- Cavazza, M.; Pizzi, D.; Charles, F.; Vogt, T.; and André, E. 2009. Emotional input for character-based interactive storytelling. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 1*, AAMAS '09, 313–320. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems.
- Chang, H.-M., and Soo, V.-W. 2009. Planning-based narrative generation in simulated game universes. *IEEE Trans. Comput. Intellig. and AI in Games* 1(3):200–213.
- Ehrig, H.; Rozenberg, G.; Kreowski, H.; and Montanari, U. 1999. *Handbook of Graph Grammars and Computing by Graph Transformation: Concurrency, parallelism, and distribution*. Handbook of Graph Grammars and Computing by Graph Transformation. World Scientific.
- Giannatos, S.; Cheong, Y.-G.; Nelson, M.; and Yannakakis, G. N. 2012. Generating narrative action schemas for suspense. In *Proceedings of the Artificial Intelligence and Interactive Digital Entertainment Conference*.
- Kybartas, B. 2013. Design and analysis of ReGEN, a narrative generation tool. Master's thesis, McGill University.
- Mateas, M., and Stern, A. 2005. Procedural authorship: A case-study of the interactive drama facade. In *Digital Arts and Culture (DAC)*.
- McCoy, J.; Treanor, M.; Samuel, B.; Tarse, B.; Mateas, M.; and Wardrip-Fruin, N. 2010. Authoring game-based interactive narrative using social games and *comme il faut*. In *Proceedings of the 4th International Conference & Festival of the Electronic Literature Organization: Archive & Innovate (ELO 2010)*.
- Peinado, F., and Gervás, P. 2006. Evaluation of automatic generation of basic stories. *New Generation Computing* 24:289–302.
- Pérez y Pérez, R.; Ortiz, O.; Luna, W.; Negrete, S.; Castellanos, V.; Peñalosa, E.; and Ávila, R. 2011. A system for evaluating novelty in computer generated narratives. In *Proceedings of the Second International Conference on Computational Creativity*, 63–68.
- Porteous, J.; Cavazza, M.; and Charles, F. 2010. Applying planning to interactive storytelling: Narrative control using state constraints. *ACM Trans. Intell. Syst. Technol.* 1(2):10:1–10:21.
- Riedl, M. O., and Young, R. M. 2010. Narrative planning: balancing plot and character. *J. Artif. Int. Res.* 39(1):217–268.
- Verbrugge, C., and Zhang, P. 2010. Analyzing computer game narratives. In *Entertainment Computing – ICEC 2010, 9th International Conference*, number 6243 in LNCS, 224–231. Springer-Verlag.
- Ware, S. G., and Young, R. M. 2012. Validating a plan-based model of narrative conflict. In *Proceedings of the International Conference on the Foundations of Digital Games*, FDG '12, 220–227. New York, NY, USA: ACM.
- Ware, S. G.; Young, R. M.; Harrison, B.; and Roberts, D. L. 2012. Four quantitative metrics describing narrative conflict. In *Proceedings of the 5th international conference on Interactive Storytelling*, ICIDS'12, 18–29. Berlin, Heidelberg: Springer-Verlag.