

Honey, I’m Home: An Adventure Game with Procedurally Generated Narrative Puzzles

Lilian Morgan¹[0000-0003-1227-5741] and Mads Haahr¹[0000-0002-9273-6458]

¹ School of Computer Science and Statistics, Trinity College Dublin, Ireland
{limorgan, haahr}@tcd.ie

Abstract. We present *Honey, I’m Home*, a short 2D adventure game which makes use of the SPHINX framework for procedurally generating narrative puzzles. The player guides the protagonist, a journalist for a local newspaper, through four game areas, interacting with numerous characters and objects along the way. The player must solve puzzles to complete each area, combining objects and interacting with characters in the game. The procedural generation of puzzles ensures that while the gameworld remains largely identical between replays, the puzzles encountered are different. The aim of *Honey, I’m Home* was to serve as a tool for our two-fold evaluation of the SPHINX algorithm, from its functionality in game development, as well as its effect on player experience. To this end a small user study was also conducted on *Honey, I’m Home*.

Keywords: Procedural Content Generation, Puzzle Games.

1 Introduction

Story Puzzle Heuristics for Interactive Narrative eXperiences (SPHINX) was developed by Barbara de Kegel and Mads Haahr [1] and aims to provide an alternative to other systems for procedural generation of narrative puzzles, such as the Puzzle-Dice system [2]. As presented at the ICIDS 2019 conference, the system is focused on narrative puzzles, i.e., puzzles that require the player to examine their surroundings, and to interact with objects and characters to progress in the story.

At the core of the SPHINX framework is an algorithm based on an extended context-free grammar. The puzzle designer provides the algorithm with three core elements: items, rules and areas. To allow for further freedom and expressivity, optional information such as properties can also be defined on items and rules. The format of the rules is given below (1), as well as an example rule using properties (2).

$$\text{Main Output [By-products]} ::= \text{Action Input}(s) \quad (1)$$

$$\text{Radio [ison:True]} ::= \text{SwitchOff Radio[ison:False]} \quad (2)$$

The puzzle generation happens at runtime on a per-area basis. Each area is associated with at least one goal, from which a form of backwards substitution creates a puzzle “tree” (see Fig. 1). This same information is also used during puzzle solution, where

the tree is traversed in the opposite direction. This method of generation helps guarantee the solvability of a puzzle, a consideration that is critical to player satisfaction. In addition to this feature, the SPHINX framework also aims to provide a large amount of expressive freedom to puzzle designers, as well as the possibility of using the algorithm across numerous game genres. As the system can function independently of core game mechanics and game worlds, it could, for example, be modified and integrated into open-world games or, potentially, into games that are already at a later stage in development. These possibilities could be explored further in future projects.

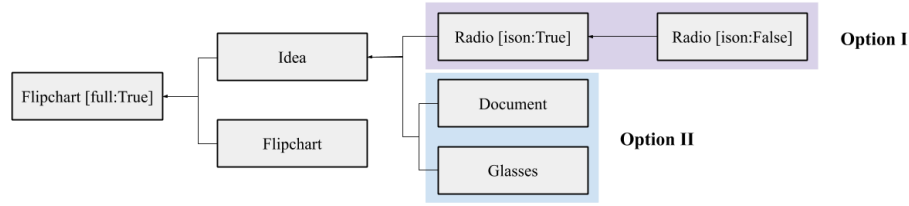


Fig. 1. An example puzzle tree from *Honey, I’m Home* showing an area with goal *Flipchart[full:True]* where the generator chooses randomly between options I and II to get the *Idea* item.

During initial conceptualization and implementation of the SPHINX framework within the Unity game engine, a small proof-of-concept game was developed. To further evaluate the overall framework, it was suggested SPHINX be integrated into another, larger game and tested within a small user study [1]. For the purpose of this two-fold evaluation, the story-based adventure game *Honey, I’m Home* was created. This demonstration aims to present how the SPHINX framework can be used to enrich aspects of gameplay, using *Honey, I’m Home* as an example.

2 Design and Development

In *Honey, I’m Home*, an atmospheric 2D side-scrolling adventure game, the overall objective for the player is to guide the protagonist, a journalist for a local newspaper, home. In order to progress in the game, the player must interact with objects and characters in four different game areas. Starting in the journalist’s *Office*, the player is asked to help disguise the fact that the day was rather unproductive. Then, in the *Landing*, an ID card needs to be acquired to be able to call the elevator and finally leave the office building. Once outside, the journalist visits the *Pub* down the road: either to find a story to submit to the newspaper or find a present to bring home. After having completed one of these tasks, all that remains is for the player to get home – be that via the park, a car or a taxi in the *Street* (see Fig. 2).

The intended time to finish *Honey, I’m Home* is between 10-15 minutes and is available to play online¹. On average, the puzzles associated with each game area increase

¹ *Honey, I’m Home* is hosted on simmer.io: <https://simmer.io/@honeyimhome/honey-i-m-home>

in length and difficulty as the player progresses, such that it should take the least amount of time to complete the *Office* area puzzle and the most for the *Pub* or *Street* areas. This setup aims to provide a more natural learning curve, allowing the player to get used to the game mechanics.

Aesthetically, *Honey, I'm Home* aims to create a slightly alternative art deco/mid-century world, where the odd occurrence of a UFO in the night sky is not entirely surprising. All artwork and animations were created specifically for the game. A number of audio tracks and effects were sourced to support the overall atmosphere and also provide feedback to the player. On the completion of an area, for example, an old-fashioned oven timer goes off.

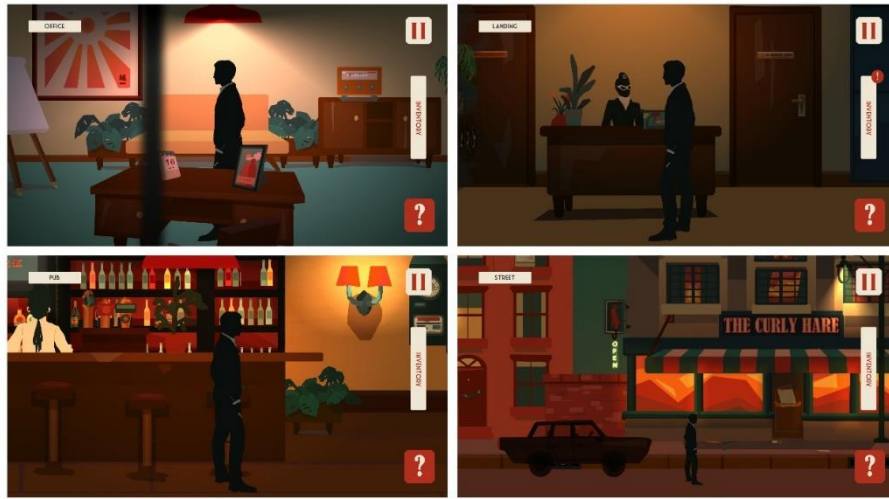


Fig. 2. Screenshots taken from *Honey, I'm Home*, showing the four different game areas; top to bottom, left to right – *Office*, *Landing*, *Pub* and *Street*.

The elements used during puzzle generation by the SPHINX system, include four game areas, 59 puzzle items and 84 puzzle rules. In addition to this, some minor modifications were made to the SPHINX framework in order to refine the experience for both the puzzle designer and the player:

- A basic “selection” system, as used in many adventure or puzzle games such as the *Rusty Lake* game series [4], was added such that the player needs to select specific items from their inventory to interact with others.
- The *Player* puzzle item was also designated to be used as a progress tracker, when needed. This is useful when the result of an action does not necessarily logically correspond to a directly interactable item.
- Certain items in the game world react when properties are changed following player interaction. This may affect the appearance of the game object, play or stop localized music (e.g., on switching on/off the *Radio*) or give access to other areas.

- A transcript for reviewing dialogues – or monologues – can be found in the pause menu, as inspired by *L.A. Noire* [3].
- A simple randomizer allows for a game object to have a list of prefabs, one of which is then chosen randomly and spawned at runtime. For example, the cars on the street make use of this function and differ in colour, depending on the random prefab selected.

While the overall level of variation in puzzles generated from one playthrough of *Honey, I'm Home* to the next is successful, the algorithm's random choice relies heavily on the breadth of the grammar provided by the puzzle designer [1]. Unfortunately, this does not guarantee maximal puzzle variation, resulting in some of the same puzzles being chosen in successive playthroughs, despite other options being available within the grammar. In an attempt to lessen this effect, a basic memory system was recently added to keep track of rules chosen in a playthrough. The next time a puzzle is then generated for an area, the generator will, if possible, select different rules. To further be able to objectively evaluate the degree of variation in the produced puzzles, the puzzle for each area is saved to a text file during runtime.

3 Conclusion

The focus of the overall project was to put the SPHINX approach to procedural generation of narrative puzzles to the test. The resulting creation, *Honey, I'm Home*, demonstrates that SPHINX can be used to add complexity and range to puzzles. To further support this, the results from a small user study conducted online were generally very positive. Players felt that the variation in puzzles from one playthrough to the next contributed positively to the overall game experience and resulted in improved re-playability of the game, as well as a better understanding of the game world and mechanics. Approaching future possibilities, participants of the user study also showed a lot of interest in the addition of procedural generation for narrative elements in larger games. As the general concept of the SPHINX system allows for modifications and adaptations to the needs of specific games, the potential for integration across genre boundaries is promising.

References

1. De Kegel, B., Haahr, M.: Towards Procedural Generation of Narrative Puzzles for Adventure Games. In: Proceedings 12th International Conference on Interactive Digital Storytelling, pp. 243-249. Springer, Snowbird, Utah, USA (2019).
2. Fernández-Vara, C., Thomson, A.: Procedural generation of narrative puzzles in adventure games: the puzzle-dice system. In: Proceedings of the Third Workshop on Procedural Content Generation in Games, p. 12. ACM (2012)
3. Team Bondi, Rockstar Games: *L.A. Noire* (2011).
4. Rusty Lake B.V. Ex.: *Rusty Lake Series* (2015-2019).