Distributed Episode Control System for Interactive Narrative Entertainment

Jun'ichi Hoshino, Katsutoki Hamana, Shiratori Kazuto, and Atsushi Nakano

University of Tsukuba, Graduate school of Systems and Information Engineering,
1-1-1, Tennodai, Tsukuba-shi, Ibaraki, Japan
{hamana, nakano}@entcomp.esys.tsukuba.ac.jp, jhoshino@esys.tsukuba.ac.jp

Abstract. We propose the massive action control system (MACS) for interactive narrative entertainment. MACS determines the action priorities for characters based in part on their own internal states, such as the motivation behind the action, feeling, and personality. MACS selects a behavior control module, called an episode tree, of about 1000 events, which is divided into action types based on these internal states and external situations. We demonstrate the effectiveness of the system with the Spilant World interactive animation contents at the National Museum of Emerging Science and Innovation in Japan, and NAMCO amusement park.

Keywords: narrative entertainment, massive action, episode tree, lifelike

1 Introduction

In recent years, many entertainment systems have relied on the progress of interaction technology to create characters that act autonomously. To show these lifelike characters, it is important for them to perform various actions, such as daily actions, reflex actions that require reacting to input from a user, perceiving actions where the character perceives an object and reacts to it, and actions based on personalities or feelings. This results in the problem of complex action planning. A character has to carry out the actions listed, keep schedules and maintain a personality, and react flexibly to user interaction, while still maintaining story flow.

In this paper, we propose the massive action control system (MACS), which can execute various actions in multiple characters (Fig. 1). This system continuously selects an appropriate fragmentary behavior control module, called an episode tree, based on the character's inner states, such as motives, feelings, and personality, and the state of the external world, such as other characters and objects surrounding the character. Thus, the character can also respond to freely timed user interference. It has three main features:

- **Structuring Actions:** Conditional branching becomes very complex when the control system treats individual actions that are elements of units, such as standing, sitting, or waving. Thus, we define each action as one structure (episode tree), where multiple elements and the start or end conditions are combined hierarchically.

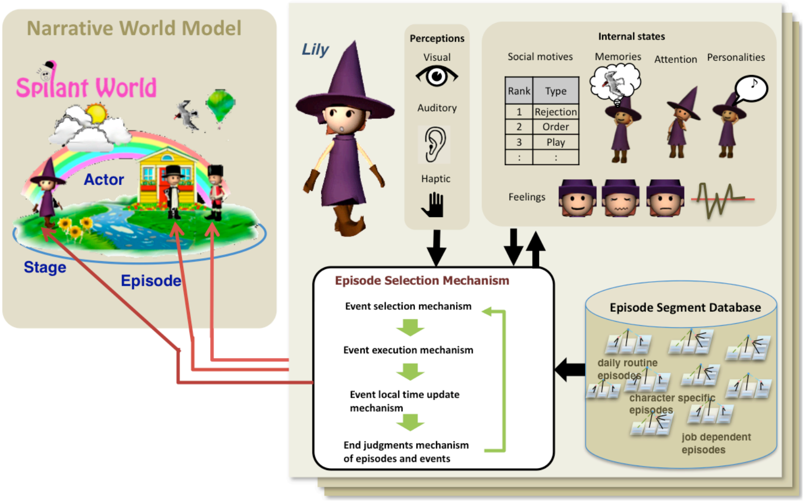

Fig. 1. MACS system configuration

Fig. 2. Spilant World narrative entertainment system at NAMCO amusement park. Operating everyday from September 2008.

- **Prioritizing Actions:** A character's intentions are important in entertainment systems. Therefore, our system prioritizes actions based on internal states, such as the character's personality, significant motivations behind the actions, and the character's feelings at that time.
- **Interpolating Actions:** If a character's action sequence changes without notice, the action will not seem natural. Therefore, it is necessary to create transitional actions to link a sequence of actions smoothly together. Pre-processing before each episode tree node assists in transitioning between actions and returning to an action performed in the past. Post-processing cleans out previous actions at the end of an episode and when current actions are interrupted. MACS interpolates the actions by inserting special processes automatically.

We created an interactive entertainment system called Spilant World (Fig. 2) to demonstrate the action control system. Multiple characters have motivations behind their actions, feelings, and personalities, and live daily in this application.

2 Related Work

One of the problems in story-based interactive entertainment is action planning. Characters have schedules such as work, meals, and physiological phenomena, as well as unique parameters such as memories and feelings, like humans in the real world. The SIMS 2, produced by Electronic Arts (EA), and Kenran Buto Sai, produced by Sony Computer Entertainment Inc. (SCEI), are typical examples of such games. Tale-Spin [13] generates a story from facts in the virtual world and rules governing the behavior of the characters. Improv [17] can produce an actor that initiates reactions in the user or other actors by a script in real time. The Oz-Project [11] proposes an interactive drama in which speech develops from interaction between the user and a globular character that has eyes, nose, and a mouth, and displays feelings. Other approaches include controlling characters' actions using the Cognitive Modeling Language (CML), which can intuitively impart knowledge to the character about an action and the preconditions influencing it [7], story generation with dynamic planning of affiliate of individual action in Dual Dijkstra's search for planning [14], and controlling characters' actions by describing a story arc that uses Hierarchical Task Networks (HTN), i.e., a task tree arranged hierarchically [4]. Façade is proposed as an interactive drama in which the game progresses by natural-language conversation as a more reformative work [12].

Furthermore, investigations of systems where the user can interact with animal characters have also been advanced. Live World [20] is a system where the user can make an object and give life to it, with the altered object behaving like an animal. Dobie T. Coyote [1] and Alpha Wolf [19] are applications that use reactions to interaction with the user. As for the former, the user can feel the vitality by seeing the dog character that learns in the user's training by the remote control training actually used. The latter is an application that reproduces social actions of a wolf; the user can see lifelike animation through reproduction of the wolf's reaction based on the actions of the wolfpack. The user can interfere in the wolf's activities by barking into the microphone.

3 Massive Action Control System

3.1 MACS Mechanisms

To construct lifelike animated characters, it is important to have them perform various actions, such as daily tasks, reflexes, acts based on their perceptions, and actions based on personalities or feelings. The MACS performs these various actions in complex narrative situations. Each character uses the system (Fig. 1). The inputs to MACS are sensory information from the external narrative world, the character's current actions, and its internal status. MACS outputs various actions based on the input information and the stored episode group. Episodes in which multiple characters interact are shared by the characters.

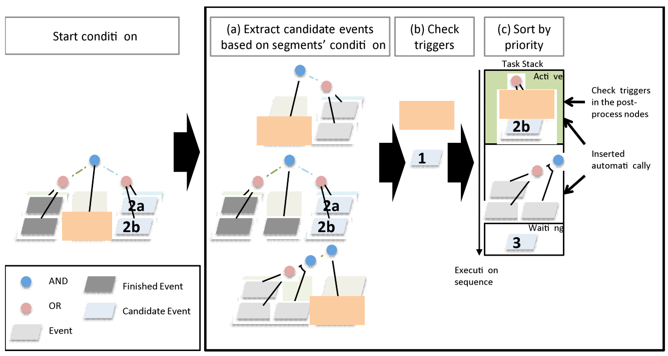

Fig. 3. Event selection mechanism

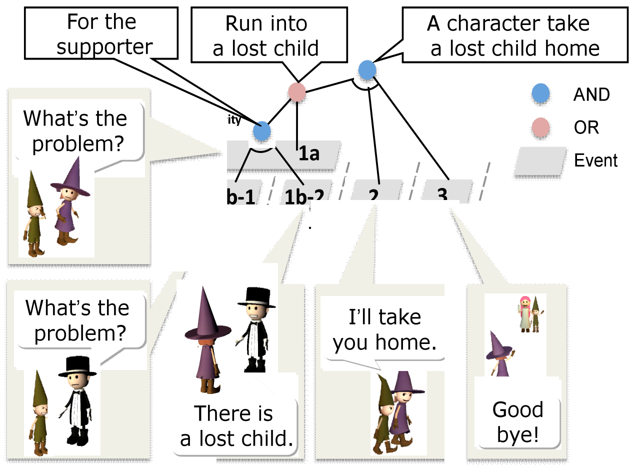

Fig. 4. Episode tree and Event contents

This system consists of four mechanisms of event selection, event execution, updating event time, and end judgments for episodes/events (Fig. 1).

- **Event selection mechanism:** This mechanism extracts events satisfying AND/OR conditions of a higher rank class from the episode tree under execution and selects an event having the highest priority in the episode. As illustrated in Fig. 3, the event selection process is as follows.

  1 Extracting candidates for the next event considering a recently finished event (a).
  2 Adding candidates to task lists if conditions in an event are satisfied (b).
  3 Sorting tasks in order of the priority (c).
  4 Selecting the event having the highest priority as the next event (c).

- **Event Execution mechanism:** This mechanism performs all actions included in the selected event. An event has a local timeline, which begins when the event starts. The characters perform the appropriate action corresponding to a moment in local time.

- **Event Local Time Update mechanism:** This mechanism updates local time in events.
- **End Judgment mechanism for episodes and events:** This mechanism judges the end of episode trees and events.

By repeating and processing these mechanisms, each character selects and executes events. However, when multiple characters share the same talk like the narrative as an exception, these characters share the same event and perform it synchronously.

3.3 Action Structures and Prioritizing Actions

As described in Figure 4, an episode tree has AND/OR nodes in its higher rank class. The AND node becomes true after all the events of the child nodes are performed and finished. The OR node becomes true after at least one of the child nodes is performed and finished. The flow of an episode is made easy to see by expressing its abstract contents by the nodes in the higher rank class of its tree. The events are arranged in the lowermost part of the tree. An event is a structure with a trigger and an action, and when the conditions used as a trigger are fulfilled, it performs the specified action. A rightward story line expresses the story's direction of movement, which advances by changing the events chosen as candidates, judging the AND/OR conditions specified as the node of the tree.

To create lifelike characters, the character's actions must be prioritized based on its personality. MACS assigns priority to actions based on Murray's 20 social action motives [15]. Each episode tree is a set of actions, and so it refers to one of these motives. For each character, MACS also ranks the motives to which that character should give priority, and compares the ranking of motives when executing an action. Even when the episode tree fulfills conditions based on situations in the existing external world. MACS selects an episode tree close to the action motive of a character using these priorities, and thus gives individuality to a character.

In the proposed system, reflex actions and daily actions are treated exceptionally in order to make a character lifelike. Reflex actions are performed without thinking, and their priority is always the highest. In this system, they are actions that occur when a character is touched or grasped. Daily actions show daily life. Even when the character perceives nothing, it is not lifelike unless it carries out its regular actions. We then prepare an episode tree of daily actions that have the lowest priority in the episode tree group, and even when not fulfilling any particular conditions, the character follows its routine.

3.4 Action Interpolation

This section describes an action interpolation function. The pre-processing and post-processing tasks corresponding to each event can be arranged as nodes of an episode tree. When moving from the current event to the next event, the action is appropriately interpolated by inserting the following processing.

1. Confirm whether the post-processing node has been placed at the node of the higher rank class of the event that has been selected now.
2. If so, add it to the task list as a top-priority task in post-processing.
3. Confirm whether the pre-processing node has been placed to a higher rank class of the selected event.
4. If so, add it to the task list as a top-priority task if the post-processing is not inserted as the task of executing it as follows if it has been inserted.
5. Execute tasks in the following order: the post-processing task, the pre-processing task added to the task list, and the event that is selected and executed next.

The following two cases can occur during the interpolation processing.

A) The event selected next changes when the post-processing task is executed.

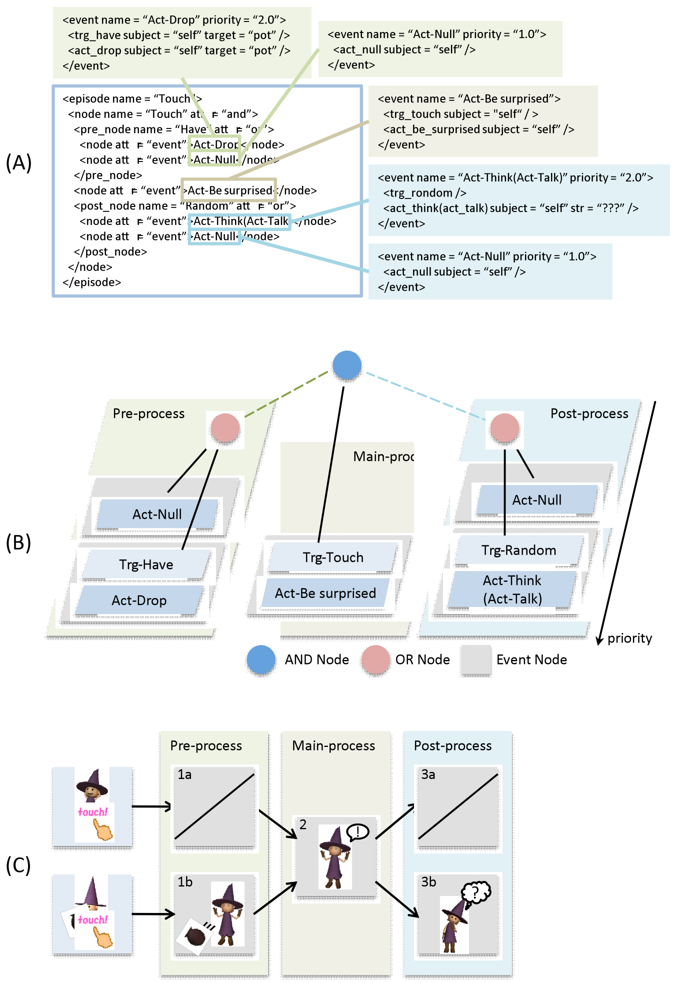

Fig. 5. (A) Scripting episode tree using XML. (B) Illustrating the script: episode tree. (C) Character’s action by the script: reflex action.

B) The event selected next changes when the pre-processing task is executed.

The processing when these two cases arise is as follows. In Case A, all pre-processing tasks stacked to the task list are deleted, but the post-processing task is certainly executed because it is a resolution of the preceding event. Alternatively, if the pre-processing tasks exists in a higher rank class of the event that will be newly selected and executed next, they are added to the task list. In Case B, if the post-processing tasks exist in a higher rank class than the pre-processing task that will be executed next, they are added, and the action to schedule for execution in the interim is determined. In addition, if the pre-processing tasks exist in a higher rank class of the event that will be newly executed, they are added to the task list.

4 Spilant World Narrative Entertainment System

This section describes an interactive application called Spilant World in which a user can drag an icon in the narrative world and also appreciate the various character's reactions by touching or grasping the object in the virtual world. An early version of the application is presented in [16]. In this paper, we extend the Spilant World system using MACS, and demonstrate the effectiveness of the system at the National Museum of Emerging Science and Innovation in Japan, and NAMCO amusement park.

4.1 System Overview

This system consists of a 37' LCD equipped with an optical touch panel, speaker, and PC. A virtual world is displayed, and the user can participate in the world by placing a finger on the touch panel. The screen display is as follows. The virtual world has a vivid display with the icons for object addition located along the right-hand side. The user can perform four operations: touching an object or character with a finger; creating a new object by holding an icon and dragging and dropping it in the virtual world; grasping a character or object by touching for an extended time; and drawing a line by touching the screen and moving the hand quickly. When performing a direct interaction on a character via touching/grasping, the user can see the character's reaction. Moreover, the character's reaction can be seen when the user interacts indirectly by adding an object and drawing an event line. (See Fig. 6.)

4.2 Number of Episodes and Aliveness

We experimented following two cases and compared it. Case 1 is when the number of episode trees is 321 and Case 2 is when it is 946. However, in practice, since it is removed from the candidate for search by the motive of a character, the former is 100, and the latter is 390. Comparison of action of the character about those two cases is shown in Fig. 9. In (A), the user added a bloom object. In (B), the user grasped and moved a character. And in (C), the user touched a light. The horizontal axis is progress of time.

(A) In Case 1, a character reacted to a broom, said “It’s a broom! I’ll take it.”, and approached it. Then, she took it and only walked. And, another character didn’t react to her with a broom. It compares, in Case 2, the occurrence until she speaks is the same as a Case 1, but, she flew using the broom. Then, another character was surprised at the sight.

(B) In Case 1, when the grasped character was moved, there was some characters were surprised it and there was also some characters were not surprised it. It compares, in Case 2, almost all the characters that are present in the neighborhood show the reaction.

(C) In Case 1 and in Case 2, the light had be lighting similarly first. Then, the blue character both indicated the reaction “It shines!” However, after that, in Case 1, only he showed interest and other characters had not looked at all. It compares, in Case 2, they said “Beautiful!” and “Oh!” Each character reacts to the light.

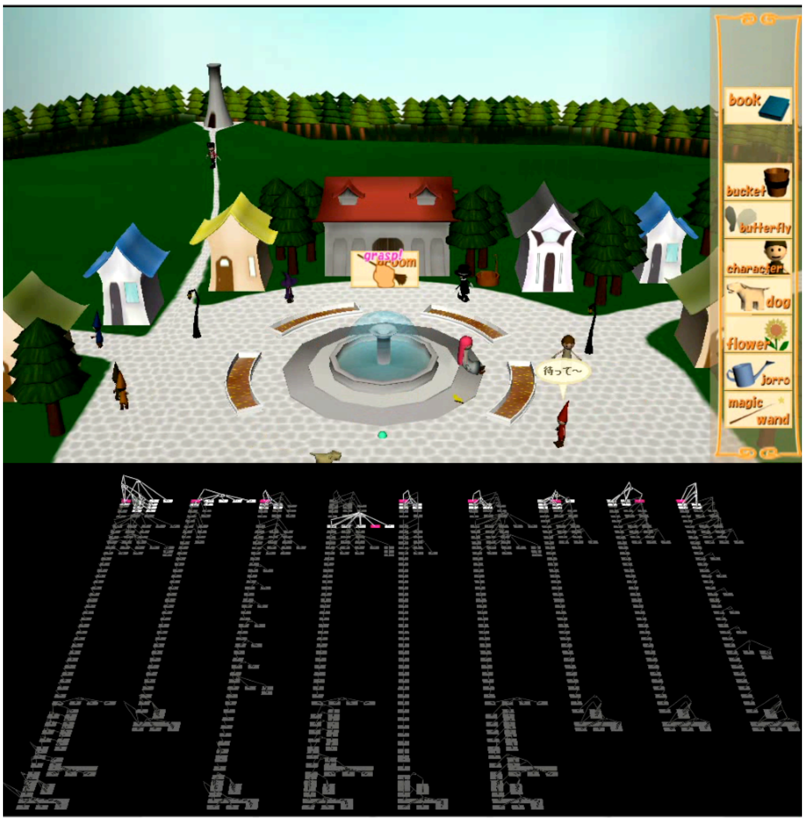

Fig. 6. A screen capture from Spilant World system with internal episode status. Active episode trees are highlighted. Red modules are currently executed by the characters.

Fig. 7. Many children have experience our Spilant World system at the National Museum of Emerging Science and Innovation in Japan

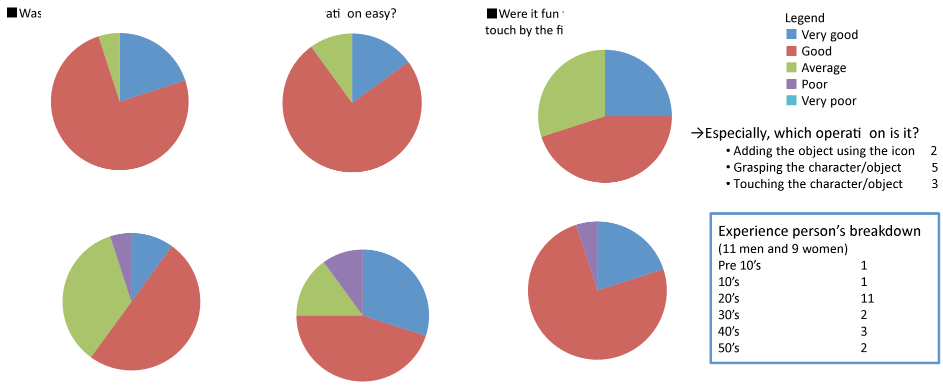

Fig. 8. Questionnaire result

Table 2. The number of episode trees and events

Character	The number of episode trees	The number of events
Lily	25	107
Toto	21	77
John	21	67
Sala	24	80
Scot	7	22
Tom	4	13
Dog	13	29
Others	13	39
Total	128	434

Table 3. The number of times and the frequency of appearance of events

Order	Event	The number of times	The frequency
0	WalkAround1	548	14.96
1	WalkAround2	329	8.98
2	grasp_event1	234	6.39
3	grasp_event2	230	6.28
4	WalkAround3	119	3.25
5	WalkAround4	99	2.7
6	fall_fountain_event3	97	2.65
7	on_the_roof1	93	2.54
8	fall_fountain_event2	80	2.18
9	on_the_roof2	74	2.02
10	on_the_roof3	67	1.83
11	find_broom_pre	54	1.47
12	find_butterfly1	54	1.47
13	find_butterfly_post	52	1.42
14	on_the_grass1	50	1.37

4.3 Demonstration

Fig. 7 illustrates the situation at the National Museum of Emerging Science and Innovation in Japan. Many children have experienced our system. Then, we recorded the user's operation and the state of event and episode selected by the character/object. The number of episode trees is 128 and the number of events is 434. The items for every object are as in Table 2.

4.3.1 Evaluation by Record

Record of the event/episode for every character/object is described. As an example, top 15 pieces are shown in Table 3 about the number of times of an event appearance of a character called "Lily". The event which consecutive numbers attach in the table is an event in the same episode, and shows the appearance order. As shown also in this table, schedule action of WalkAround (event which takes a walk) etc. has appeared mostly, and it is because the character was always acting along with the schedule if the user weren't interference to the character. grasp_event (event held) and fall_fountain_event (event dropped on a fountain) have appeared a lot next. These are events which happen when the user interferences to a character, i.e., reflex actions are had appeared a lot. Many find_ (event which finds something) have appeared continuously, and these are perceiving actions of a character. About active action, because many events were not able to be created, it didn't appear a lot. This result was the same in general about other characters.

In addition, we think, on reaction chaining, the character acted continuously and the system display more lifelike character. When record was investigated, about Lily, about 17% was the chin to the reaction from reaction. And, a lot of chains comparatively are the chains related to grasp.

4.3.2 Evaluation by Questionnaire

The questionnaire was also able to be taken from 20 persons who experienced the application. (See Fig. 8.) Most persons think that the operation is easy and many persons think that the operation itself is fun. Therefore it is thought that there was no problem about operation. When seeing it without interaction, the number of the persons who were able to feel the character lifelike was half. If the character finishes reacting to some extent, new reaction will not be generated any more. Therefore, a possibility that the character is only taking a walk is high. The user was seeing it and replies that a character would not be lifelike. When operating, it is increasing because the character certainly react to the user interference.

5 Conclusion

In this paper, we proposed MACS, which can show characters autonomously carrying out various reactions to interference from the user in an interactive entertainment application. MACS expresses and accumulates various actions of a character with similar structures (episode trees). Thus, MACS can perform massive actions by controlling them systematically. Our experiment and exhibition demonstrated the results.

There is two directivity of scalability. The first is to make the episode tree dynamic, because a dynamic episode tree can create new storylines. The other purpose is to prepare tools to edit the episode trees visually. Because the trees are scripted by XML, we can edit them easily. However, we believe that editing will be easier if we prepare a technical graphical user interfaces.

References

1. Blumberg, M, B. 2002. D Learning: What Dog Learning tells us about Building Characters that can Learn. Chapter from Exploring Artificial Intelligence in the new Millennium, ed. G. Lakemeyer & B. Nobel, Morgan Kaufmann Publishers.
2. Buss, A. H. 1971. Aggression pays. In I. J. L. Singer (ed.) The control of aggression and violence. Academic Press.
3. Cassell, J., Vilhjálmsson, H. and Bickmore, T. 2001. BEAT: the Behavior Expression Animation Toolkit. In Proceedings of SIGGRAPH 01, 477-486.
4. Cavazza, M., Charles, F. and Mead, S.J., 2001. AI-based Animation for Interactive Storytelling. Proceedings of Computer Animation, IEEE Computer Society Press, Seoul, Korea, pp. 113-120.
5. Ekman, P. and Friesen, W. V. 1975. Unmasking the Face. Prentice-Hall.
6. Emmons, R. A. 1989. The personal striving approach to personality. In L.A. Pervin (Ed.), Goal concept in personality and social psychology. Lawrence Erlbaum. 87-126.
7. Funge, J., Tu, X. and Terzopoulos, D. 1999. Cognitive modeling: knowledge, reasoning and planning for intelligent characters. In Proceedings of SIGGRAPH 99, 29-38.
8. Kopp, S. and Wachsmuth, I. 2004. Synthesizing Multimodal Utterances for Conversational Agents, The Journal Computer Animation and Virtual Worlds 15(1), 39-52.
9. Maes, P., Darrell, T., Blumberg, B. and Pentland, A. 1995 The ALIVE system: full-body interaction with autonomous agents, Computer Animation, 11-18.
10. Masuko, S. and Hoshino, J. 2007. Head-eye animation corresponding to a conversation for CG characters. Computer Graphics Forum: Journal of the European Association for Computer Graphics,vol.26, no.3, 303-311.
11. Mateas, M. 1997. An Oz-centric review of interactive drama and believable agents. Technical Report CMU-CS-97-156, School of Computer Science. Pittsburgh, PA: Carnegie Mellon University.
12. Mateas, M. and Stern, A. 2003. Façade: An Experiment in Building a Fully-Realized Interactive Drama, In Game Developer's Conference: Game Design Track, San Jose, California
13. Meehan, J. 1976. The metanovel: Writing stories by computer. Ph.D. Dissertation. Yale University.
14. Mori, H. and Hoshino, J. 2005. Key Action Technique for Digital Storytelling, IFIP 4th International Conference on Entertainment Computing, 2005. 36-47.
15. Murray, E. J. 1964 Motivation and emotion. Englewood Clifts, New Jersey: Prentice Hall.

16. Nakano, A., Koumura, J., Miura, E. and Hoshino, J. Spilant World: Interactive Emergent Story Game using Episode Tree, The Journal of the Society for Art and Sciece, Vol. 6 No. 3, pp.145-153, 2007, Japan.
17. Perlin, K. and Goldberg, A. 1996. Improv: A System for Scripting Interactive Actors in Virtual Worlds. In Proceedings of SIGGRAPH 96, 205-216.
18. Stone, M., Decarlo, D., Oh, I., Rodriguez, C., Stere, A., Lees , A. and Bregler, C. 2004. Speaking with hands: creating animated conversational characters from recordings of human performance, ACM Transactions on Graphics 23(3), 506-513.
19. Tomlinson, B., Downie, M., Berlin, M., Gray, J., Lyons, D, Cochran, J. and Blumberg, B. 2002. Leashing the AlphaWolves: mixing user direction with autonomous emotion in a pack of semi-autonomous virtual character. In Proceedings of SIGGRAPH 02, 7-14.
20. Travers, M. 1994. Recursive Interfaces for Reactive Objects. Proceedings of the SIGCHI conference on Human factors in computing systems, 379-385.

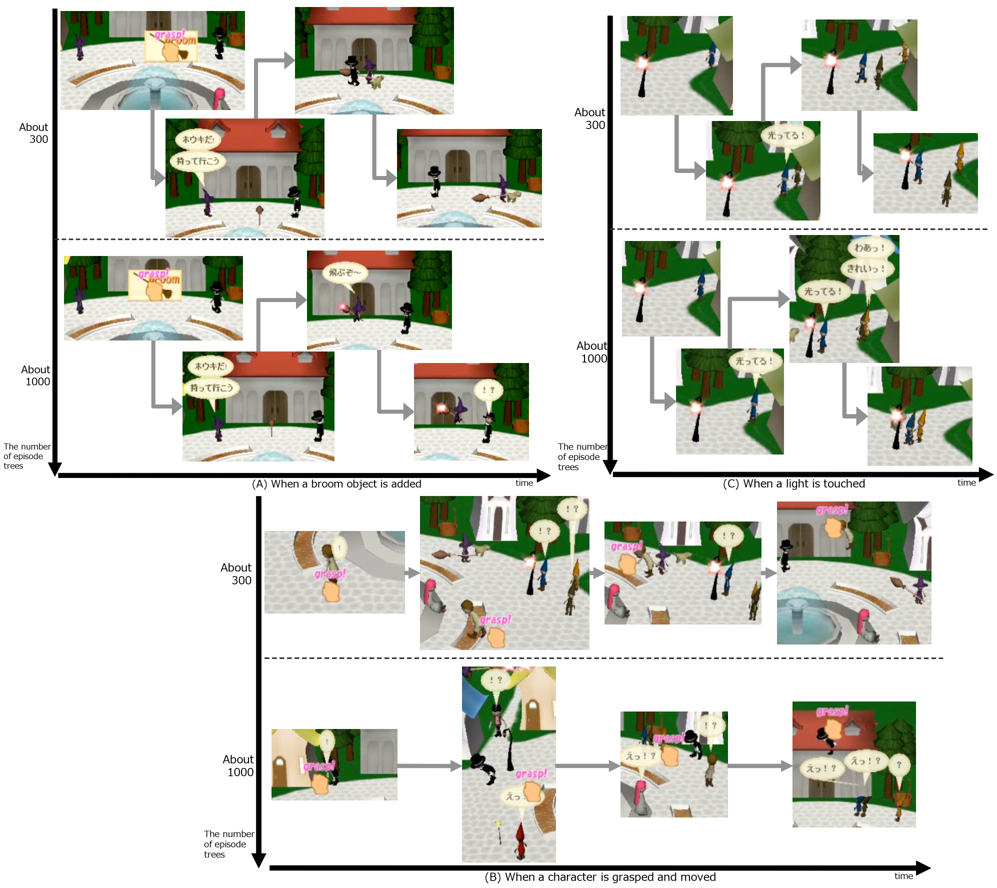

Fig. 9. Comparison of the number of episode trees, and the reactions of the characters.