Frame Selection Using Iterative Grammatical Evolution for Automatic Comic Generation from Game Logs

Ruck Thawonmas, *Senior Member, IEEE* and Yoshinori Tani

Abstract— **Recently, we developed a system that visualizes a game play, from its game log, with a comic-style representation. In that system, comic frames are selected based on their interaction levels without considering users' interests. In this paper, we tackle this issue by proposing an interactive frame-selection method that adopts interactive evolutionary computation (IEC) for evolving conditional rules used in selection of frames. IEC enables the user to directly give feedbacks on which an originally proposed subjective fitness function bases. In order to lessen user fatigue, the proposed method evolves the frame-selection conditional rules for certain consecutive generations based on this subjective fitness function – taking into account the latest user feedback – and another originally proposed objective fitness function. The objective fitness function ensures the consistency in the ratio among event types in a resulting comic with the ratio in the game log. As a promising evolutionary computation technique for the task in this work, we use grammatical evolution. We then evaluate the proposed method by a user evaluation using typical game logs. Experimental results confirm that evaluation participants favor comics whose frames are selected by the proposed method more than comics generated by a baseline method.**

I. Introduction

Our recently developed comic-generation system summarizes online-game experiences with a comic-like representation [1], [2]. This kind of system enables its users (game players) to look back their plays or share their experiences with other players. Hence, it promotes communication and strengthens social relations between players. This feature also enables player retention in the game [3], a crucial issue in online gaming.

In the above system, comic frames are selected based on the level of interactions. Namely, frames are selected from a set of frame candidates in decreasing order of the number of actions occurring therein. Although a different weight can be assigned to each action for representing its importance, the system is not equipped with a mechanism for the user to give a feedback about intermediate comics. For example, a player of interest may favor frames containing a fight with monster A than that with monster B. This paper presents our solution to this issue. Our solution is based on interactive evolutionary computation, which takes into account user feedbacks during evolution.

However, the more feedbacks, the more fatigue the user must bear because she or he has to let the system know which frames they favor and which they do not for each generated comic. How to mitigate user fatigue is another challenging research issue. in order to solve this issue, we adopt and modify a method introduced by Quiroz et al. [4] for evaluating user interfaces. The method allows evolution for certain consecutive generations after receiving a feedback by the user. The contributions of this paper are as follows:

1) In Section III, we describe our comic-generation system that allows for generation of comics favored by the user due to incorporation of user feedbacks in selection of comic frames.
2) In Section IV, we present a grammar for deriving frame-selection conditional rules from genomes evolved by grammatical evolution (GE); this grammar permits selection of each frame based on a combination of conditions.
3) In Section V, we describe an interactive frame-selection method that reduces user fatigue due to the advent of a proposed fitness function that enables GE to evolve genomes for certain consecutive generations with each user feedback.

We point out here that our approach is applicable to generation of comics for any game or online game whose game engine is accessible, for rendering comic frames.

II. Related Work on Use of Comic-Style Representation

To our knowledge, there was no previous work that focuses on comic-frame selection in an interactive fashion based on GE. Other existing work on comic generation from gaming experiences includes Chan et al. [5] and Shamir et al [6]; their frame selection methods are also based on the interaction level as done in the aforementioned authors' previous work [1], [2]. The comic-style representation was also used for summarization of experiences in a conference [7], daily experiences [8], videos or movies [9]–[11], as well as museum-visit experiences [12]–[14].

III. Outline of the Comic Generation System

Figure 1 depicts an overview of our comic generation system that includes the proposed interactive frame-selection module, replacing the frame selection module in our previous system [1]. As done in [1], first, a game log of interest is chronologically divided into multiple partitions, each called a scene consisting of multiple events. Derivation of scenes and events therein are based on the level of interactions. An event is considered as a frame candidate and categorized into five categories (henceforth called idioms): **new scene**, **talk**, **shoot**, **approach**, and **other**. These idioms represent situations where the main player character (PC) enters a

The authors are with the Intelligent Computer Entertainment Laboratory, Graduate School of Science and Engineering, Ritsumeikan University, Shiga, Japan (phone: +81-77-561-5048; fax: +81-77-561-5203; email: ruck@ci.ritsumei.ac.jp).

978-1-4577-0011-8/11/$26.00 ©2011 IEEE

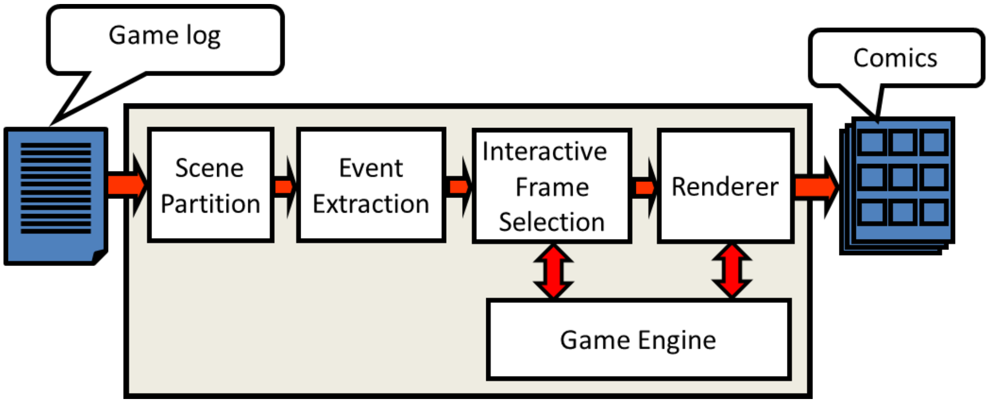


Fig. 1. Overview of the comic generation system

new map, talks with other PCs or non-player characters (NPCs), shoots them, encounters them, and conducts other actions, respectively. An idiom is decided for each frame candidate based on the residing actions and their occurrence frequencies. After frames have been selected from the set of frame candidates by the proposed interactive frame-selection method, parameters for deciding frame size and shape are decided. These parameters are used together with the camera angle, camera position, and zoom position for rendering comic frames.

IV. Grammatical Evolution for Frame Selection

GE [15] uses a variable length linear genome representation for representing programs. The genome is an integer array whose elements are called codons. Typical genetic operators are applied to evolve such a genome. The process for mapping a genotype to phenotype is conducted based on a user-specified grammar in Backus-Naur Form (BNF). We use GE in this work, rather than more wildly known genetic algorithm or genetic programming, because GE provides more flexibility in selection of programing languages and more simplicity in implementation of complex functions. Recently, a system was proposed by Shao et al. [16] that uses an interactive version of GE for interactive generative music.

Figure 2 shows our grammar for frame selection; where ⟨rule⟩, ⟨expr⟩, and ⟨condition⟩ are non-terminals; ⟨setup⟩, ⟨ifs⟩, and ⟨operator⟩ are terminals; cond(x) is a function that returns 1 when x is true and otherwise 0. Each of the resulting ⟨condition⟩s is used as the conditional rule for selecting the corresponding frame candidate. Supposing that there are M frame candidates, in chronological order, in the frame-candidate list and that the number of resulting ⟨condition⟩s is N, we give the frame-selection algorithm as follows:

```
1: $i = 0$
2: **while** $i < N$ and $i < M$ **do**
3: if the $i$th ⟨condition⟩ holds,
4:     then select the $i$th the frame candidate
5: increment $i$
6:**end while**
```

Now we describe how the genotype-phenotype mapping process works. Note that the 1st ⟨setup⟩ (npc), the 2nd ⟨setup⟩ ($target$), the 3rd ⟨setup⟩ ($idiom$), ⟨expr⟩, ⟨condition⟩, ⟨ifs⟩, and ⟨operator⟩ have 2, 2, 5, 3, 3, 3, and 2 choices, respectively. GE uses the following function – Rule = c mod r – to determine which choices are selected for the phenotype, where c is the codon integer value and r is the number of choices for the current symbol.

For example, given the following genome of five codons 34, 11, 29, 40, and 80; the first three codons are used for the setup of npc, $target$, and $idiom$, respectively. This results in selection of 0th (34 mod 2), 1st (11 mod 2), and 4th (29 mod 5) choices, respectively, i.e., $npc = 0$, $target = 1$, and $idiom$ = **other**. The forth codon is used for deciding ⟨expr⟩, resulting to selection of ⟨condition⟩ ⟨rule⟩. The fifth codon is then used for deciding the ⟨cond⟩, resulting to selection of ⟨ifs⟩.

Now, the end of the genome has been reached, but there are still nonterminals (⟨ifs⟩ and ⟨rule⟩) left in the phenotype. In this case, the first codon is used again for deciding ⟨ifs⟩. This leads to selection of "$target$ == cond(the camera target is an NPC)", which becomes the conditional rule for the 0th frame candidate. The second codon is used again for deciding the setup of npc. This process continues until no non-terminals remain or the number of derived selection rules (N) is equal to the number of frame candidates (M). Eventually, the following two conditional rules for frame selection are obtained.

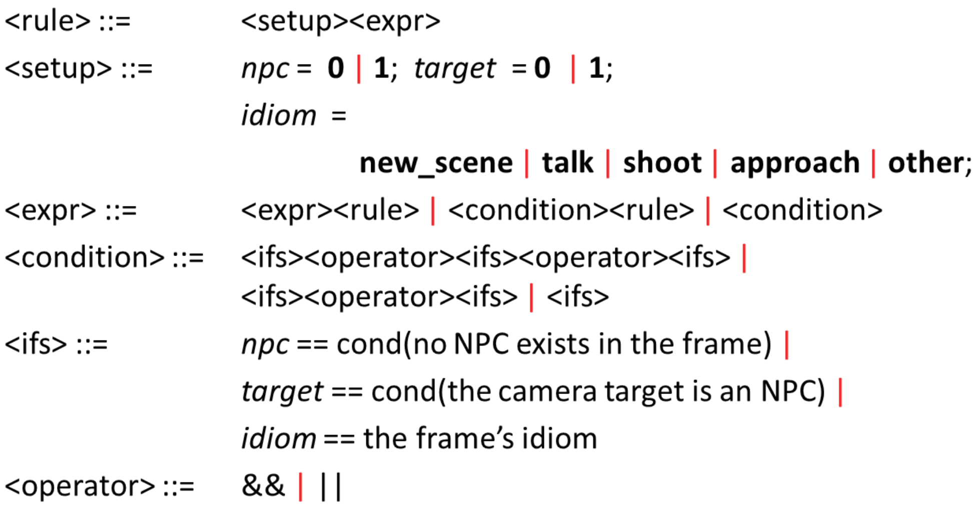
Rule 0: "1 == cond(the camera target is an NPC)"

Rule 1: "$idiom$ == the frame's idiom ||
1 == cond(the camera target is an NPC)"

Figure 3 summarizes this mapping process.

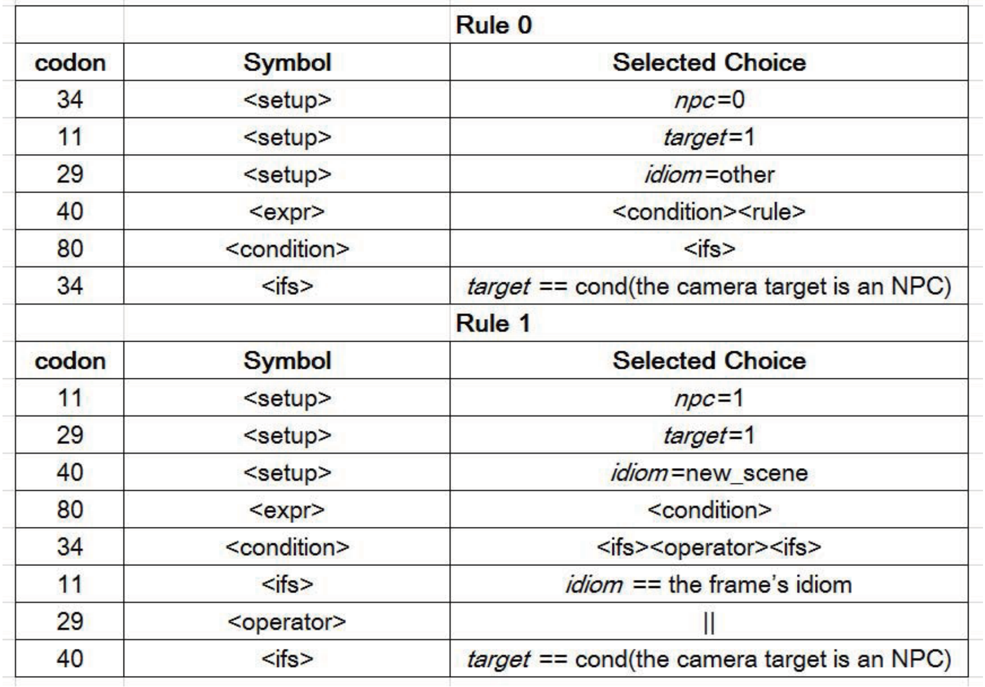
V. Interactive Frame Selection

The user interacts with the comic generation system via viewing and giving a feedback about a set of generated intermediate comics. For the evolution process, we propose a comic fitness function that consists of a subjective fitness



```
<rule> ::=        <setup><expr>
<setup> ::=       npc = 0 | 1; target = 0 | 1;
                  idiom =
                        new_scene | talk | shoot | approach | other;
<expr> ::=        <expr><rule> | <condition><rule> | <condition>
<condition> ::=   <ifs><operator><ifs><operator><ifs> |
                  <ifs><operator><ifs> | <ifs>
<ifs> ::=         npc == cond(no NPC exists in the frame) |
                  target == cond(the camera target is an NPC) |
                  idiom == the frame's idiom
<operator> ::=    && | ||
```

Fig. 2. Grammar used in frame selection



		Rule 0
codon	Symbol	Selected Choice
34	<setup>	*npc*=0
11	<setup>	*target*=1
29	<setup>	*idiom*=other
40	<expr>	<condition><rule>
80	<condition>	<ifs>
34	<ifs>	*target* == cond(the camera target is an NPC)
		Rule 1
codon	Symbol	Selected Choice
11	<setup>	*npc*=1
29	<setup>	*target*=1
40	<setup>	*idiom*=new_scene
80	<expr>	<condition>
34	<condition>	<ifs><operator><ifs>
11	<ifs>	*idiom* == the frame's idiom
29	<operator>	||
40	<ifs>	*target* == cond(the camera target is an NPC)

Fig. 3. Genotype to phenotype mapping process from a genome consisting of five codons: 34, 11, 29, 40, and 80

function and an objective fitness function. User feedbacks are used in computation of the former function.

To receive a user feedback, our system presents C comics, individually generated from each of the best C genomes, and then it asks the user to select the best comic and the worst comic. We consider that genomes leading to comics similar to the best comic should be rewarded while those similar to the worst comic be penalized. Thereby, we define the aforementioned subjective fitness function for comic i, $sub(i)$, as follows:

$$sub(i) = d(worst, i) - d(best, i), \tag{1}$$

where $d(x, y)$ represents the distance between comic x and comic y. For a distance metric between a pair of comics, we empirically consider that comics are similar if their ratios among idioms are similar. Hence, $d(i, j)$ is defined as follows:

$$\begin{aligned} d(i,j) = &|new\_scene(i) - new\_scene(j)| + \\ &|talk(i) - talk(j)| + |shoot(i) - shoot(j)| + \\ &|approach(i) - approach(j)| + \\ &|other(i) - other(j)|, \end{aligned} \tag{2}$$

where $idiom(x)$ represents the ratio of the corresponding idiom in all frames of comic x.

In order to well represent the user experience in a given game log, we consider that a comic should have the ratio among idioms similar to that of the game log. Following the same recipe in (2), given $\#new\_scene$, $\#talk$, $\#shoot$, $\#approach$, and $\#other$, the ratios in the game log of idioms new scene, talk, shoot, approach, and other, respectively, we define the aforementioned objective fitness function of comic i, $obj(i)$, as follows:

$$\begin{aligned} obj(i) = {} & |new\_scene(i) - \#new\_scene| + \\ & |talk(i) - \#talk| + |shoot(i) - \#shoot| + \\ & |approach(i) - \#approach| + \\ & |other(i) - \#other| \end{aligned} \tag{3}$$

The fitness function of comic i (or genome i whose phenotype leads to the frame selection rules of this comic), $fitness(i)$, is given below as

$$fitness(i) = sub(i) - obj(i) \tag{4}$$

The procedure that we use for interactive frame selection is as follows:

Step 1: Initialize the genome population of P genomes.

Step 2: Generate C comics from the best C genomes, present them to the user, and receive a feedback from the user.

Step 3: Use GE to evolve the population for E generations based on the fitness function in (4); at each generation the following genetic operators are conducted.

- single-point crossover
- mutation
- selection of the best P genomes from both parents and their children

Step 4: If the termination condition does not hold, goto Step 2.

Step 5: Generate the final comic whose frames are selected according to the phenotype of the best genome.

VI. Evaluation

We compared comics generated by the proposed frame-selection method and a baseline frame-selection method. The latter method selects comic frames according to their interaction levels [1]. We used same camera-work rules for them. As in our previous work, because dialogs play an important role to the narration of resulting comics, we implemented these two methods in our comic-generation system such that they always select frames with the talk idiom.

We first requested 12 participants, 3rd year students in our department, to watch a video clip [1] showing a typical online-game play by a player. In this play, the player character first met a mission master who gave a mission to annihilate a tyrannosaurus at another map, and, during the journey from the first map to the destination map, the player character also fought with a wolf. The game log of this player, lasting about six minutes, was used in comic generation. After watching the video clip, each participant was asked to participate in an interactive frame-selection session and then to compare two comics, whose frames were generated by the proposed method and the baseline method with the same number of frames.

TABLE I
Average score of each method for each of the six questions

Question #	Proposed Method	Baseline Method
Q1	3.42	2.5
Q2	4.25	3.25
Q3	3	2.17
Q4	3.17	2.17
Q5	3.5	2.67
Q6	3.58	2.75

As for GE related settings, a gnome consists of 20 codons, each having an integer value within the range [0, 200] and being randomly initialized within this range. Other parameters are as follows: P = 5, C = 5, E = 5, crossover probability = 0.5, mutation probability = 0.1. If a codon is selected for mutation, its value will be randomly assigned an integer value within the range [0, 200]. Our termination condition is to stop after the 25th generation.

Each participant was asked to make selection every five generations (E = 5). During comic comparison, each participant was shown two comics, without being told which method was used for a given comic, and asked to answer the following six questions for each comic in a typical five-level Likert scale:

Q1 Does the comic have an interesting story?

Q2 Does the comic have its content well representing the video-clip play?

Q3 Does the comic have an easy-to-watch camera-work?

Q4 Does the comic have the minimum amount of unnecessary frames?

Q5 Does the comic have a dynamic representation?

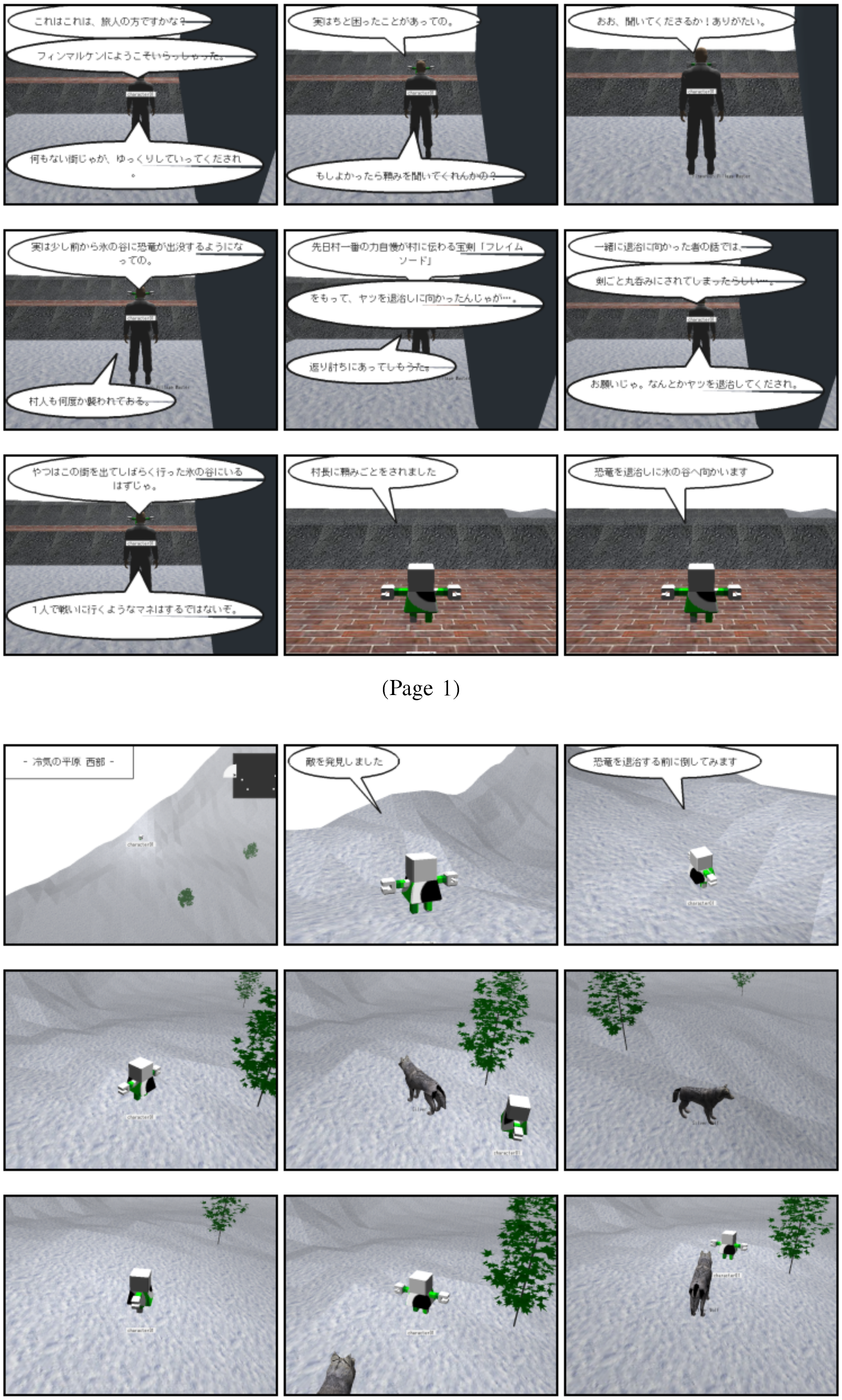
Q6 Does the comic summarize the play according to your favor?

The participants were also asked to write a short reason behind their rating for each question.

Table I shows the average score of each method for each of the six questions. The proposed method outperforms the baseline in all questions. We conducted the t-test analysis and found that the differences between the two methods for all questions are statistically significant, i.e., $p < 0.05$. However, the average score of the proposed method is 3 for Q3, which implies that the camerawork of the resulting comics should be improved.

Figures 4 and 5 show the first two pages of the comic generated with the proposed method by one of the 12 participants and those of the comic generated with the baseline method, respectively. English translations of the residing dialogs and map names are given in Fig. 6 and 7, respectively.

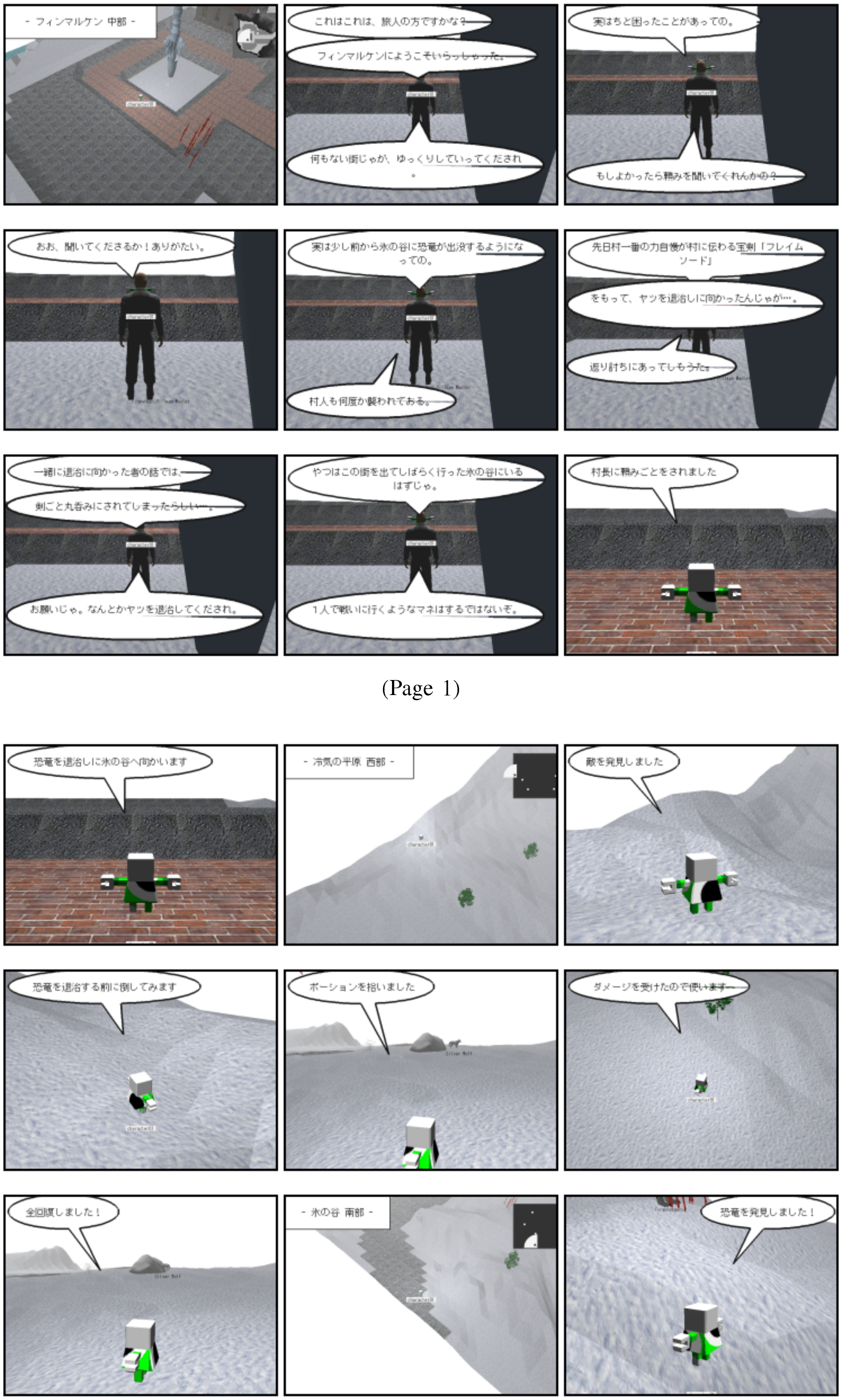
[1] http://www.youtube.com/watch_popup?v=SOsBInZOcs8&vq=medium


これはこれは、旅人の方ですかな？
フィンマルケンにようこそいらっしゃった。
何もない街じゃが、ゆっくりしていってくだされ。
実はちと困ったことがあっての。
もしよかったら頼みを聞いてくれんかの？
おお、聞いてくださるか！ありがたい。
実は少し前から氷の谷に恐竜が出没するようになっての。
村人も何度か襲われておる。
先日村一番の力自慢が村に伝わる宝剣「フレイムソード」
をもって、ヤツを退治しに向かったんじゃが…。
返り討ちにあってしもうた。
一緒に退治に向かった者の話では、
剣ごと丸呑みにされてしまったらしい…。
お願いじゃ。なんとかヤツを退治してくだされ。
やつはこの街を出てしばらく行った氷の谷にいるはずじゃ。
１人で戦いに行くようなマネはするではないぞ。
村長に頼みごとをされました
恐竜を退治しに氷の谷へ向かいます
－ 冷気の平原 西部 －
敵を発見しました
恐竜を退治する前に倒してみます

(Page 1)

(Page 2)

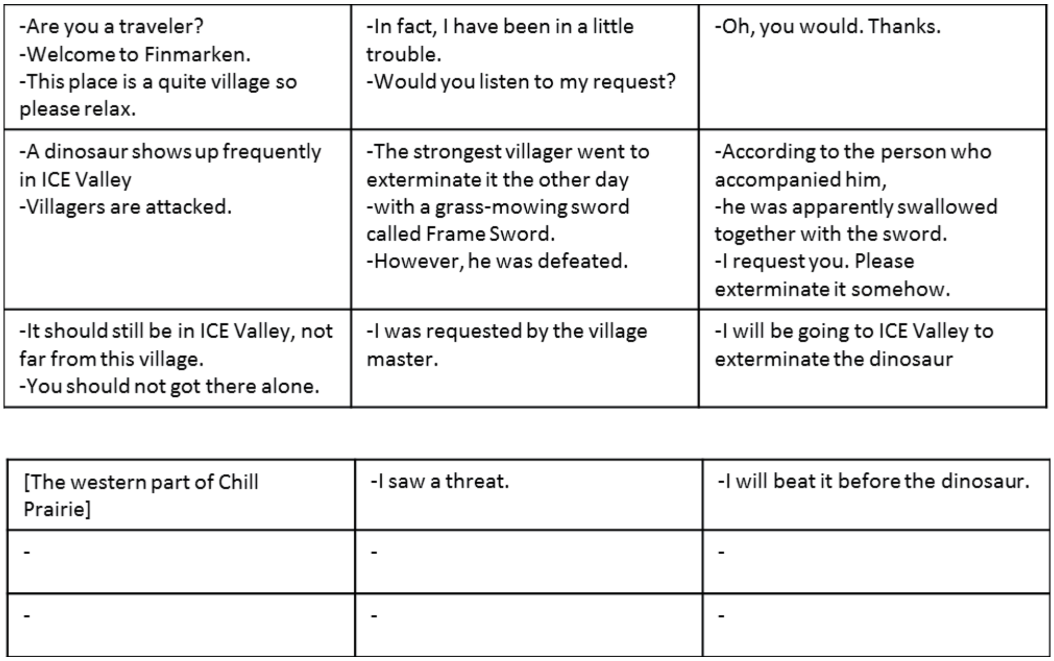
Fig. 4. Example of the first two pages of a comic whose frames are selected by the proposed interactive frame-selection method based on feedbacks from one of the 12 participants


- フィンマルケン 中部 -
これはこれは、旅人の方ですかな？
フィンマルケンにようこそいらっしゃった。
何もない街じゃが、ゆっくりしていってくだされ。
実はちと困ったことがあっての。
もしよかったら頼みを聞いてくれんかの？
おお、聞いてくださるか！ありがたい。
実は少し前から氷の谷に恐竜が出没するようになっての。
村人も何度か襲われておる。
先日村一番の力自慢が村に伝わる宝剣「フレイムソード」
をもって、ヤツを退治しに向かったんじゃが…。
返り討ちにあってしもうた。
一緒に退治に向かった者の話では、
剣ごと丸呑みにされてしまったらしい…。
お願いじゃ。なんとかヤツを退治してくだされ。
やつはこの街を出てしばらく行った氷の谷にいるはずじゃ。
１人で戦いに行くようなマネはするではないぞ。
村長に頼みごとをされました
恐竜を退治しに氷の谷へ向かいます
- 冷気の平原 西部 -
敵を発見しました
恐竜を退治する前に倒してみます
ポーションを拾いました
ダメージを受けたので使います
全回復しました！
- 氷の谷 南部 -
恐竜を発見しました！

(Page 1)

(Page 2)

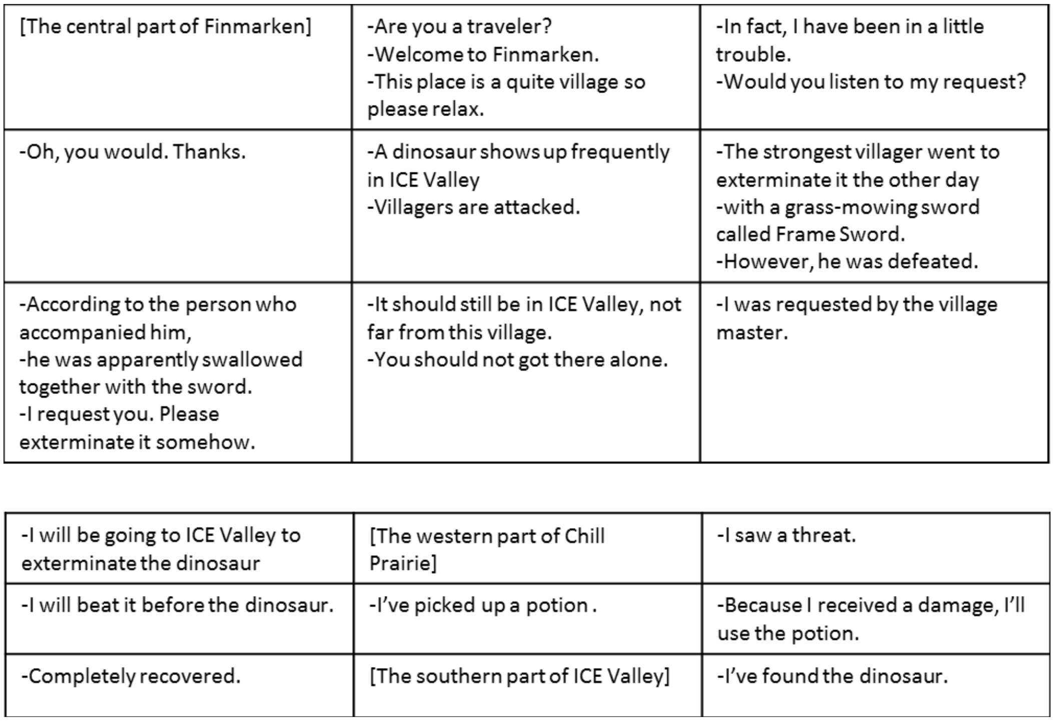
Fig. 5. Example of the first two pages of a comic whose frames are selected by the baseline method with the number of frames equal to that of the comic in Fig. 4



-Are you a traveler? -Welcome to Finmarken. -This place is a quite village so please relax.	-In fact, I have been in a little trouble. -Would you listen to my request?	-Oh, you would. Thanks.
-A dinosaur shows up frequently in ICE Valley -Villagers are attacked.	-The strongest villager went to exterminate it the other day -with a grass-mowing sword called Frame Sword. -However, he was defeated.	-According to the person who accompanied him, -he was apparently swallowed together with the sword. -I request you. Please exterminate it somehow.
-It should still be in ICE Valley, not far from this village. -You should not got there alone.	-I was requested by the village master.	-I will be going to ICE Valley to exterminate the dinosaur

[The western part of Chill Prairie]	-I saw a threat.	-I will beat it before the dinosaur.
-	-	-
-	-	-

Fig. 6. English translation of the dialogs and map names in Fig. 4



[The central part of Finmarken]	-Are you a traveler? -Welcome to Finmarken. -This place is a quite village so please relax.	-In fact, I have been in a little trouble. -Would you listen to my request?
-Oh, you would. Thanks.	-A dinosaur shows up frequently in ICE Valley -Villagers are attacked.	-The strongest villager went to exterminate it the other day -with a grass-mowing sword called Frame Sword. -However, he was defeated.
-According to the person who accompanied him, -he was apparently swallowed together with the sword. -I request you. Please exterminate it somehow.	-It should still be in ICE Valley, not far from this village. -You should not got there alone.	-I was requested by the village master.

-I will be going to ICE Valley to exterminate the dinosaur	[The western part of Chill Prairie]	-I saw a threat.
-I will beat it before the dinosaur.	-I've picked up a potion .	-Because I received a damage, I'll use the potion.
-Completely recovered.	[The southern part of ICE Valley]	-I've found the dinosaur.

Fig. 7. English translation of the dialogs and map names in Fig. 5

Both comics show in the first page the user experience in which the player character was receiving the mission by the mission master. However, in the second page, the former comic successfully shows the user experience in which the player character was fighting against the wolf during the mission journey while the latter comic does not. During the interactive frame-selection session, at each selection, this participant selected a comic that includes the aforementioned wolf as the best comic. The participant commented that one of the reasons he gave a higher rate to the former comic was because of the appearance of the wolf.

VII. Conclusions and Future Work

We presented the comic generation system for generating a comic from game log and proposed the interactive frame-selection method. This system allows the user to interact with the system through viewing intermediate comics and giving feedbacks about them. The proposed method evolves genomes, or frame-selection rules, using GE based on the proposed fitness function. Two functions construct this fitness function, i.e., the subjective fitness function taking into account user feedbacks, and the objective function keeping the consistency between the ratio among the idioms in a resulting comic and that in the game log. Due to the advent of these functions, evolution of genomes for certain consecutive generations is possible after receiving a user feedback. This reduces the user burden in giving feedbacks to the system. The conducted user evaluation indicates that the comics whose frames are selected by the proposed method are more favored than those whose frames are selected by the baseline method.

As our future work, we plan to increase the expressivity of comics by extending our use of interactive GE. One possible research theme is to strengthen our comic layout mechanism. This would increase the variety of comic frames, such as large frames, small frames, slanted frames, etc. As already stated in Sect. V, another theme is to improve the camerawork so that a more proper set of camera parameters – the camera angle, camera position, and zoom position – is applied to a given frame. These extensions will be based on our previous work focusing on comic layout [1] and camerawork control [17] in order to achieve mechanisms for interactive frame layout and interactive camerawork control, respectively.

Acknowledgments

This work is supported in part by the – MEXT Global COE Program – Digital Humanities Center for Japanese Arts and Cultures, Ritsumeikan University, and by Grant-in-Aid for Scientific Research (C), No. 23500192, the Japan Society for Promotion of Science.

References

[1] R. Thawonmas and T. Shuda, “Comic Layout for Automatic Comic Generation from Game Log,” *IFIP International Federation for Information Processing*, vol. 279/2008 (1st IFIP Entertainment Computing Symposium), pp. 105–115, 2008.

[2] T. Shuda and R. Thawonmas, “Frame Selection for Automatic Comic Generation from Game Log,” *Lecture Notes in Computer Science*, vol. 5309 (ICEC 2008: Entertainment Computing), pp. 179–184, 2008.

[3] V. Posea, M. Balint, A. Dimitriu, A. Iosup, “An analysis of the BBO fans online social gaming community,” *Roedunet International Conference (RoEduNet)*, 2010 9th, Jun 24-26, 2010.

[4] J.C. Quiroz, S.J. Louis, A. Shankar, and S.M. Dascalu, “Interactive Genetic Algorithms for User Interface Design,” *Proc. IEEE Congress on Evolutionary Computation 2007*, pp. 1366–1373, 2007.

[5] C.-J. Chan, R. Thawonmas, and K.-T. Chen, “Automatic Storytelling in Comics: A Case Study on World of Warcraft,” *CHI Extended Abstracts 2009*, pp. 3589–3594, 2009.

[6] A. Shamir, M. Rubinstein, and T. Levinboim, “Generating Comics from 3D Interactive Computer Graphics,” *IEEE Computer Graphics and Applications*, vol. 26, no. 3, pp. 53–61, 2006.

[7] Y. Sumi, R. Sakamoto, K. Nakao, and K. Mase, “ComicDiary: Representing individual experiences in a comic style,” *Lecture Notes in Computer Science*, vol. 2498/2002 (Ubicomp 2002), pp. 16–32, 2002.

[8] S.-B. Cho, K.-J. Kim, and K.-S. Hwang, “Generating Cartoon-Style Summary of Daily Life with Multimedia Mobile Devices,” *Lecture Notes in Computer Science*, vol. 4570/2007 (IEA/AIE 2007), pp. 135–144, 2007.

[9] W.-I. Hwang, P.-J. Lee, B.-K. Chun, D.-S. Ryu, and H.-G. Cho, “Cinema comics: Cartoon generation from video stream,” *Proc. of GRAPP 2006*, pp. 299–304, 2006.

[10] J. Calic, D. Gibson, and N. Campbell, “Efficient Layout of Comic-like Video Summaries,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 7, pp. 931–936, 2007.

[11] H. Tobita, “DigestManga: interactive movie summarizing through comic visualization,” *CHI. Extended Abstracts 2010*, pp. 3751–3756, 2010.

[12] R. Thawonmas and A. Fukumoto, “Frame Extraction Based on Displacement Amount for Automatic Comic Generation from Metaverse Museum Visit Log,” *4th International Conference on Intelligent Interactive Multimedia Systems and Services (KES IIMSS 2011)*, University of Piraeus, Greece, Jul. 20-22, 2011.

[13] R. Thawonmas and T. Shuda, “Frame Selection for Automatic Comic Generation from Museum Playlog in Metaverse,” *IADIS International Conference Game and Entertainment Technologies 2011 (IADIS GET 2011)*, Rome, Italy, Jul. 22-24, 2011.

[14] R. Thawonmas and K. Kato, “Camerawork for Comics Generated from Visitors’ Experiences in a Virtual Museum,” *The 10th International Conference on Entertainment Computing (ICEC 2011)*, Vancouver, Canada, Oct. 5-8, 2011.

[15] M. O’Neill and C. Ryan, Grammatical Evolution: Evolutionary Automatic Programming in a Arbitrary Language, Kluwer Academic Publishers, 2003.

[16] J. Shao, J. McDermott, M. O’Neill, and A. Brabazon, “Jive: A Generative, Interactive, Virtual, Evolutionary Music System, *Lecture Notes in Computer Science*, vol. 6025/2010 (Applications of Evolutionary Computation), pp. 341–350, 2010.

[17] R. Thawonmas and K. Oda, “Rule-Based Camerawork Controller for Automatic Comic Generation from Game Log,” *Lecture Notes in Computer Science*, vol. 6243/2010 (ICEC 2010: Entertainment Computing), pp. 326–333, 2010.