Non-Branching Interactive Comics

Edirlei Soares de Lima [1], Bruno Feijó [1], Antonio L. Furtado [1], Simone D.J. Barbosa [1], Cesar T. Pozzer [2], and Angelo E.M. Ciarlini [3]

[1] PUC-Rio – Department of Informatics, Rio de Janeiro, RJ – Brazil
{elima, bfeijo, furtado, simone}@inf.puc-rio.br
[2] UFSM – Department of Applied Computing, Santa Maria, RS – Brazil
pozzer@inf.ufsm.br
[3] UNIRIO – Department of Applied Informatics, Rio de Janeiro, RJ – Brazil
angelo.ciarlini@uniriotec.br

Abstract. Comics are a unique and classical form of storytelling. The advent of interactive narratives brings the possibility of interaction to the traditional comic books. In this paper we present a non-branching interactive comics system capable of generating dynamic interactive narratives in the format of comic books. The system allows users to interact with certain objects, and then observe the consequences of their actions in the unfolding story. We validate the proposed system with a user study conducted with 18 participants. The results indicate that such systems may indeed provide an attractive form of entertainment.

Keywords: Interactive Comics, Interactive Storytelling, Comic Book

1 Introduction

Comics are a popular form of visual storytelling, wherein juxtaposed still images are combined with text. Modern comic books emerged at the turn of the 19th century and evolved in different ways in Europe, America and Japan. In the early 19th century, the Swiss artist Rodolphe Töpffer published the book *Histoire de M. Vieux Bois*, which some consider the first "comic book" [17]. In America, the genre of superheroes has dominated the mainstream for decades, and its popularity has varied widely since the first period of popularity in the 1940s, known as the American Golden Age of comic books. The appreciation for the storytelling abilities of comics, however, was even more remarkable in Japan, where *manga* (Japanese term for comics) gained reputation both for profitable sales and diversity of genres. Nowadays, classical printed books are sharing space with digital forms of comics, such as *Marvel's The Avengers: Iron Man – Mark VII* [34]. The developers of *Webcomics*, which are comics published on a website [31], are now exploring new forms of comics, such as narrative branching structures (also known as *hypercomics*) and animated panels with sounds (also known as *motion comics*). An example of hypercomic is *Meanwhile* [30], which is a branching interactive comics where one navigates making choices and solving puzzles. Through the last ten years, many researchers have attempted to transform the classical form of comics into a new form of digital interactive content. The applications include the automatic generation of non-interactive comics [5][6][8], interactive experiences for web and mobile devices [22][30], and some interactive narratives [19][20]. However, most of the applications that include some form of interactivity are based on the concept of branching narrative structures [26], which are known in the area of interactive storytelling as having several limitations, such as the authoring complexity and the lack of story diversity. Research on interactive storytelling has been exploring the generation of interactive narratives since the 1970s and may provide the proper foundation for the creation of a new form of interactive content based on comics.

The most robust forms of interactive narratives rely on artificial intelligence techniques, such as planning [15], to dynamically generate the sequence of narrative events rather than following predefined branching points. The techniques that support the dynamic generation of stories are also useful to maintain the coherence of the entire narrative. Moreover, they support the propagation of changes introduced by the users, allowing them to effectively interact and change the unfolding stories.

Although artificial intelligence techniques can help improve the diversity of stories, they face the challenge of generating comics in real-time from a story that is not known beforehand. In branching narratives, all the possible storylines are predefined by the author, and the system is prepared to represent them in the best possible way. On the other hand, in systems based on planning techniques, stories are created by the planning algorithm, guided to some extent by the user interactions, and it is not easy to predict all the possible storylines that can emerge. These unpredictable outcomes require intelligent systems capable of adapting themselves to represent emergent narratives correctly. In the case of comics, the intelligent system must also know how to use comic language to generate, in real-time, each panel representing the narrative events. In other words, the system should also generate part of the comic art.

In this paper we explore the use of artificial intelligence techniques to blend narrative generation and interactive comics. We present and evaluate a non-branching interactive comic system capable of generating dynamic interactive narratives in the format of comic books. The system allows users to freely intervene in the stories by interacting with the scene objects, and then observe the consequences of their actions in the unfolding story. As far as we are aware, there is no other work proposing a system for non-branching interactive comics in the literature.

2 Related Work

The automatic generation of comics has been an active topic of research since Kurlander *et al.* proposed their famous Comic Chat system [1] in the nineties. Comic Chat is a system capable of automatically generating comics from chat sessions. It determines the correct poses for each character and situation through a semantic analysis of the participants' messages. Following a similar approach, Sumi *et al*. present ComicDiary [2], a system that automatically creates personal diaries in a comic style.

In the gaming context, Shamir *et al*. [3] and Shuda and Thawonmas [4] propose a system to automatically generate comics from game sessions summarizing the game events. Chan *et al*. [5] adopt a similar approach and present a system that automatically summarizes players' actions and interactions in the game World of Warcraft through comics. Pizzi *et al*. [6] use comic-like storyboards to represent game level solutions as a game design tool. Their system generates all possible solutions to a given game level using the player character as the main agent. Then, they attach elements of storyboards to the planning operators so that a complete solution generates a comic strip representing the storyboard.

Alves *et al*. [7] describe an XML-based language that semantically describes comics. They also present Comics2D, a system able to interpret the XML language and generate comic strips. In a more recent work, Alves *et al*. [8] present another system able to create comic-like summaries of agent-based stories generated by the interactive storytelling system FearNot! [9]. The system analyses the story logs, characters' emotional information to understand their actions and their importance in the story, and then selects the most important events to create comic strips.

There are some interactive experiences based on comics designed for mobile devices, like Nawlz [22], which is an adventure comic book designed for iPad that combines animation, interactivity, music, and text in a panoramic comic format. However, the story is entirely linear and user interactions don’t have any effect in the story outcome. Some major comic publishers have tried to create interactive comic books. A recent example is the already cited Marvel's The Avengers: Iron Man - Mark VII [34], which is an interactive comic book designed for mobile devices that allow users to play with some interactive elements of the scenarios, but without affecting the story.

The possibility of choosing between different story paths is explored by Goodbrey [19][20]. The author presents several web applications that combine the concept of branching narratives with the idea of “infinite canvas” proposed by McCloud [21]. The commercial product Meanwhile for iPad and iPhone [30] is another example of branching technique. Taking a different approach, Andrews *et al*. [23] explore the application of interactive comics in a projected multi-touch interface. Their system projects on a sheet of paper predefined lower-level narrative elements, such as characters and dialogues, allowing users to enrich the story by adding “top level” objects.

Our system differs from the aforementioned works because we integrate three important features: plot generation; interaction affecting the plot generation; and automatic layout generation based on the plot. Furthermore, the story unfolds from a logical framework rather than from a predefined branching structure. Such an integrated and flexible approach is possible because the system is supported by the following components: (1) a planning module that automatically generates coherent and diversified stories according to the user interventions; (2) intelligent algorithms capable of generating comics in real-time. To the best of our knowledge, this is the first work on non-branching interactive comics in the literature.

3 Comics

Comics are a narrative form that uses a sequence of panels containing signs and images combined with texts, where the reader perceives time spatially. The texts are in the form of speech balloons, captions, and onomatopoeic words for sound effects (SFX). Our treatment of comics relies on the understanding that comics consist of: "sequential art", as pointed by Eisner [18]; semiotic entities, as claimed by O'Neil [29]; and juxtaposed images in deliberate sequences, where time and space are one and the same, as observed by McCloud [17]. The question of time is clearly explained by McCloud using a comparison with movies in his book. He points out that each successive movie frame is projected on exactly the same space, while each comic frame occupies a different space – and concludes that space does for comics what time does for film.

An easy way to understand the language of comics is by looking at the structure of a comic book. A comic book usually comprises one or more pages, each page consisting of panels, whose varying size and location influence the reader's progress. These panels convey space and time and behave as picture planes through which the reader views the action of the story. Inside the panels, story events are represented through three types of visual elements: figurative, iconic, and textual [16]. Figurative elements are the characters themselves, their actions, facial expressions, and gestures, representing what they are doing and what they are feeling. Iconic elements include speech balloons, thought balloons, SFX balloons, and narration boxes. These icons inform the reader which words are being thought, said, or narrated. In addition, the shapes of these icons can be altered to add nuance to the words found within them. Finally, the textual elements represent the text found within the speech balloons, thought balloons and narration boxes. Figure 1 indicates the basic elements of comic books. A more complete and detailed analysis of the visual language of comics is presented by McCloud [17] and Eisner [18].

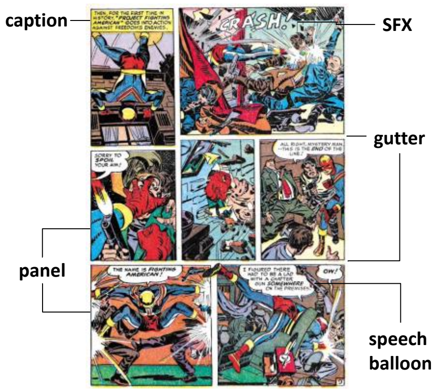


Fig. 1. Elements of comic books (panel, gutter, caption, speech balloon, SFX). Copyright material under "fair use" policy.

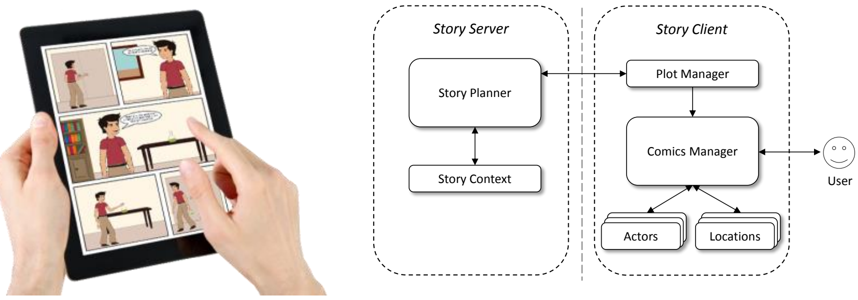
Although comics are sometimes seen as a childish form of storytelling, with images that are often deliberately simplified, they are still capable of evoking strong emotional reactions from readers, creating identification, and effectively conveying a story in an very appealing manner [3].

4 Interactive Comics

Our method to create a new form of interactive content for comics has three main tasks: (1) the automatic generation of interactive narratives using a story planner; (2) the definition and compositing of panels in the format of comic books; (3) the interaction with the user. The proposed system is embedded in a tablet computer, where users are able to visualize narratives through comic panels and interact with certain objects that can affect the unfolding stories (Figure 2a).

Initially, the system generates an entire story plot and users can read it as a traditional comic book. Additionally, some scenarios include interactive objects that can be activated by users by tapping on the tablet. When this happens, the logical context of the story is modified at that specific point of the narrative according to the effects of the activated object. The intervention propagates to the next story events, and the comic panels are updated to represent the new storyline. Users thus have a way to interact with the story and observe the consequences of their actions in real time.

Tablet computers are the most suitable platform to support the visualization of comics in a digital format, because they come closer to real comic books than desktop computers. However, the limited processing power of tablets may not be capable of running a complex planning algorithm in real time. Consequently, the proposed system is based on a client-server architecture (Figure 2b), where the server hosts the planner responsible for generating the stories, and the client contains the visualization interface that presents the narrative in the format of comics.



(a) Tablet (b) System Architecture

Fig. 2. (a) A tablet presenting an interactive comic book. (b) Architecture of the system

In the proposed architecture, the Story Planner consists of a heuristic search planner that performs a forward search in the space of world states to find a path from a given state to a goal state based on the Story Context. The Story Context defines the characters, locations, a set of authorial goals, and a set of planning operators using the STRIPS formalism [24]. On the client side, the Plot Manager is responsible for requesting new plans for the remote Story Planner through a TCP/IP connection. The Plot Manager sends the current state of the story to the Story Planner, and receives back the sequence of story events to achieve an authorial goal. Then the Plot Manager sends the story events to the Comics Manager, which uses its graphical resources

(Actors and Locations) to generate comic panels to represent the events in the format of a comic book and interacts with the user.

5 Story Planner

Interactive storytelling systems can follow three basic approaches: plot-based [10], character-based [11], or a hybrid approach [12]. In our system we adopted a plot-based approach, where the story plot is automatically built and updated in real time by a planning algorithm. The Story Planner is based on a standard Heuristic Search Planning (HSP) [13] algorithm that performs a forward search in the space of world states using a weighted A* algorithm [14] with inadmissible heuristic values. The planner solves STRIPS-like planning problems with ground operators. The Story Context contains the definition of the planning problem, which includes a set of propositions P used to describe the states and operators, a set of planning operators O, the initial state $I \subseteq P$, and a hierarchy of authorial goals $G \subseteq P$. Each operator $o \in O$ has a list of preconditions and a list of effects.

The initial state is represented by a set of propositions describing the starting point of the story. It includes the definition of the characters, their current situation, and a description of the world. Examples of propositions for the initial state could be: *character(emma)* for "Emma is a character"; *at(emma, house)* for "Emma is at House"; *healthy(emma)* for "Emma is healthy"; and *path(house, hospital)* for "There is a path connecting the House to the Hospital".

Goal states are also represented by sets of propositions, and are used by the comic writer to guide the development of the story towards an admissible outcome. The planner adopts a hierarchy of authorial goals, in the sense that if a higher goal cannot be achieved, the planner tries its successor. The planner can fail to achieve a desired goal either if there is no valid sequence of actions that leads from the initial state to the goal state; or if the prescribed time limit for searching for a solution is exceeded. In both cases, the planner tries to achieve the next successor goal from the authorial goal hierarchy.

The planning operators correspond to the events that may occur in narratives. They are represented using a STRIPS-like formalism. Each operator has: (1) a list of arguments, indicating the roles played by the characters involved in the event, and other features of the corresponding actions; (2) a list of preconditions, specifying facts that must hold prior to the execution of the event; and (3) a list of effects, specifying facts that hold immediately after the execution of the event. An example of event is "*a zombie attacking a victim*" (where $\wedge$ means AND and $\neg$ means NOT):

operator: attack(CH1, CH2, PL)
preconditions: zombie(CH1) $\wedge$ $\neg$zombie(CH2) $\wedge$ at(CH1, PL) $\wedge$ at(CH2, PL)
effects: $\neg$healthy(CH2) $\wedge$ infected(CH2)

In this case, the operator has 3 arguments: two characters (CH_1 and CH_2) and a place PL. The preconditions for the execution of this event indicate that CH_1 must be a zombie, CH_2 cannot be a zombie, and both CH_1 and CH_2 must be at the place *PL*. The effects of this event indicate that CH_2 is not healthy and is infected.

Each planning operator is also associated with an action template that describes the story event in terms of actions. For example, if *Get*(*Emma*, *Toy*, *House*) is an instance of the planning operator *Get*(*CH*, *OB*, *PL*), the event will be decomposed into 5 actions: (1) Emma looks at the object; (2) Emma says "*Wow! It's a Toy!*"; (3) Emma is happy; (4) Emma crouches; and (5) Emma gets the Toy. The use of the action templates allows the planner to produce more detailed stories without sacrificing its performance. However, it requires an addition authorial effort, since it is up to the author to specify a detailed vision of how the event occurs.

The client-server architecture of the system allows several clients to be connected with the planning server simultaneously. However, a large number of clients requesting new story plots and the server having to run several instances of the planning algorithm may compromise the real-time performance of the system. To remedy this problem, the story planner stores all computed story plots in a database of plans indexed by the initial state and the goal state. When a client requests a new story plot, the server checks the database of plans. If the requested initial state and goal state are found, the server returns the corresponding precomputed plan; only if they are not found, the server runs a new instance of the planning algorithm.

6 Panel Definition and Compositing

The proposed process for the visualization of interactive comics is divided in two phases. The panel definition phase comprises the process of assigning the story events to their corresponding panels, computing the size required for each panel, and defining the layout of each page. The panel compositing phase includes the process of rendering the content of the panels (background, characters, objects, and word balloons).

6.1 Panel Definition

Panels are used to present a single moment frozen in time. Letting *i* refer to time, a panel P_i represents a discrete instant t_i (Figure 3). Besides the association with time, the specification of panel P_i comprises a specific location L_i and a set of events E_i: $P_i = \{L_i, E_i\}, E_i = \{e_{i,1}, e_{i,2}, \cdots, e_{i,j}, \cdots, e_{i,NE}\}$. An event *e* is an instance of a planning operator (*e.g. Get(Emma, Toy, House)*) or of a simple action (*e.g. Talk(Emma, "Wow! It's a Toy"),* which corresponds to a speech balloon). Events are always sequential in time (*i.e.* the story planner does not generate parallel events), but this sequence is compressed in the discrete instant of time t_i represented by the panel P_i. The story planner generates a long sequence of events to be assigned to panels. Furthermore, the nature and emotional intensity of the events attributed to a panel determine its size and shape.

We establish the following rules to decide whether or not a new event e_n can be grouped with its preceding event e_p in a panel P, without breaking the continuity of time and space:

1. If e_n and e_p are both speeches of the same character, or different characters that are at the same place, and the number of speeches already added to the panel P is smaller than α (maximum number of speeches supported by a single panel), then the event e_n can be assigned to the panel P.
2. If e_n is a speech and e_p contains an action performed by the speaking character, then the event e_n can be assigned to the panel P.
3. If e_p and e_n are the same event (which is not a speech) performed by two different characters at the same place, then the event e_n can be assigned to the panel P.
4. Otherwise, e_n is assigned to a new panel.

The next step is the process of creating the pages that will support the panels. A page is composed of a sequence of panels with varying size and location, which present the story events to the reader. The size of a panel is generally proportional to the amount of narrative content presented, and its position is relative to the chronological order of the events. In order to dynamically calculate the size of the panels, we propose a method to estimate the importance of a panel based on weights associated with the events and the location where the events take place.

Firstly, each class of event (*e.g. Go*, *Talk*, *LookAt*) and the locations where the events can happen are associated with a weight based on their importance to the narrative. For example, a *Go* event (where a character goes from one place to another) may have less importance to the narrative than an *Attack* event (where a character is attacked and infected by a zombie). The same idea is valid to the locations where the events happen, some places are more important than others. The story author assigns weights to the events and locations. These assignments can be done by a single numerical value or a conditional expression (*e.g.* a *Go* event may have its weight increased if certain specific events occur at time t_i represented by panel P_i). Therefore the weights are calculated by a function that depends on P_i. Weights are also calculated for each row and, finally, for the whole page. We propose the following equations to calculate the weights of a panel P_i, a row R_k, and a page F_j:

$$P_i^{weight} = L_i^{weight}(P_i) + \sum_{j=1}^{NE} e_{i,j}^{weight}(P_i)$$

$$R_k^{weight} = \sum_{j=1}^{NK} P_j^{weight} \;, P_j \subseteq R_k$$

$$R_k^{weight} \leq \beta$$

$$F_j^{weight} = \sum_{k=1}^{NR} R_k^{weight} \;, NR \leq \gamma$$

where $L_i^{weight}(P_i)$ is a function that returns the weight of the location $L_i \subset P_i$; $e_{i,j}^{weight}(P_i)$ is a function that returns the weight of the event $e_{i,j} \in E_i$, $E_i \subset P_i$; NE is the number of events in the panel P_i; NK is the number of panels in a row R_k; NR is the number of rows in a page; β is the maximum value allowed for a row; and γ is the maximum number of rows allowed in a page.

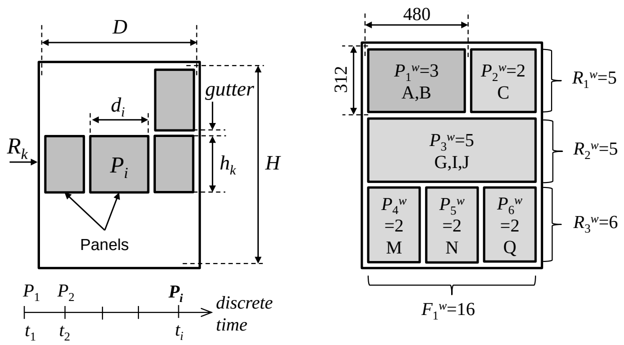
The algorithm that calculates the size of the panels and the layout of the pages starts by iterating through the panels and assigning them to a page and a row according to their chronological order. When the weight of a row (sum of the row panels' weights) reaches β (maximum weight allowed to a row), the panels begin to be assigned to next row. When the number of rows reaches the maximum number of rows per page (γ), the panels begin to be assigned to the first row of the next page. The algorithm ends when all panels are assigned to a page and a row. We must notice that the parameters β and γ determine the general aspect of the comic book page. In our prototype we assume β=6, and γ=3.

In the next step, the actual size of each panel is calculated according to its weight and position in the page. The width d_i of a panel P_i in a row R_k and the height h_i of a row R_k in a page are given by:

$$d_i(P_i, R_k) = \left(\frac{P_i^{weight}}{R_k^{weight}}\right) \times D - gutter,\ P_i \subseteq R_k$$

$$h_k(R_k, F_j) = \left(\frac{R_k^{weight}}{F_j^{weight}}\right) \times H - gutter$$

where D is the horizontal size of the page and H is the vertical size of the page in pixels, and *gutter* is the space between panels (Figure 3a). Figure 3b illustrates the process of the panel definition phase. First, the story events {A,B,C,G,I,J,M,N,Q} are grouped in panels $\{P_1, P_2, \ldots, P_6\}$ and their respective weights are calculated. Then, the panels are chronologically assigned to their corresponding rows $\{R_1, R_2, R_3\}$ and pages $\{F_1\}$, considering the maximum weight supported by a row β=6. The weights of the rows and pages are also calculated. Finally, the actual size of the panels is calculated and the layout of the page is defined. In this example, the page size is D = 800 and H = 1000 pixels, the width of the panel P_1 is $d_1(P_1, R_1)$ = 800x3/5 = 480 and its height is $h_1(R_1, F_1)$ = 1000x5/16 = 312 without discounting the gutters.,



(a) Page of a comic book (b) Example of weights

Fig. 3. (a) Elements of a comic book page of size D x H, where the panel P_i is within row R_k and has a size of d_i x h_k. (b) Example where the story events {A,B,C,G,I,J,M,N,Q} are grouped in panels {P1,…,P6}and row weights R_k^w are calculated. The subscript w stands for *weight*.

6.2 Panel Compositing

Panels can be composed of four types of objects: background layers, characters, interactive objects, and word balloons.

Background layers are a representation of the environment where the events occur. Every available location of the story context is associated with a set of static or dynamic image layers used to create the scenarios of the story. Each scenario has a set of waypoints indicating the available positions where characters and other objects can be placed during the panel compositing process. Each waypoint includes additional information about the size and angle of the character occupying that position.

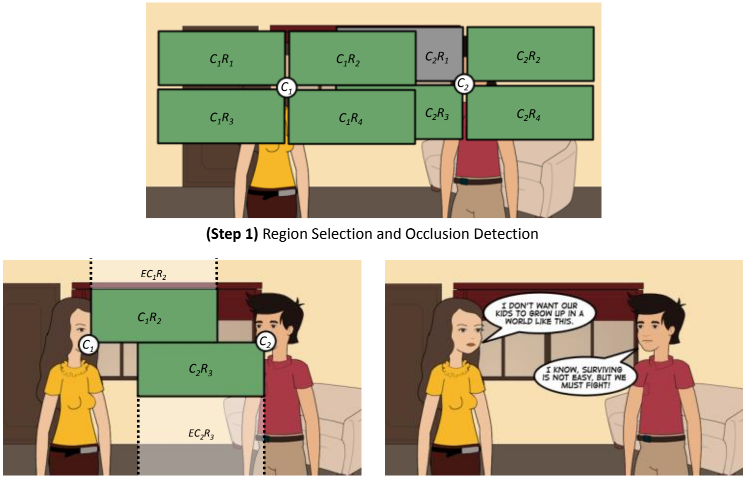
Panels are composed according to the story events that must be represented on each panel. Characters participating in the events are placed on the available waypoints of the scenario using the spatial information provided by those waypoints. The characters are composed of a set of behaviors representing the actions that they can perform during the story. Each behavior comprises a set of static images representing the action from 6 different angles. During the compositing process, the behavior is selected according to the action performed by the character, and the angle is defined by its current waypoint and action.

Interactive objects allow users to interact with the narrative by changing the content of the panels, and consequently, altering the logical context of the story. Such interventions may force the characters to take other actions and change the course of the narrative. The story author defines the interactive objects and associates them with the scenarios. Each object is composed of two images: one representing the object before user interaction, and other representing the object after its activation. During the compositing process, the objects are added to the panels as part of the scenarios.

Word balloons are the last objects to be incorporated into some panels. The correct placement of each word balloon is crucial to convey the story events without breaking the temporal continuity of the narrative. Comic writers have developed through the years some general rules on how to place word balloons in comic panels and keep the narrative content easily understandable. Based on the comic literature, we establish the following rules to adequately place a balloon B_i in its corresponding panel:

1. $B_i \cap B_n = \emptyset$: Balloons should not overlap each other.
2. $B_i \cap C_n = \emptyset$: Every balloon B_i should not overlap any of the characters C_n.
3. $B_i \cap O_n = \emptyset$: Every balloon B_i should not overlap any of the interactive objects O_n.
4. B_i must be placed according to its chronological and reading order.

In order to comply with these rules, we adopted a method for balloon placement based on the procedure presented by Chun *et al.* [25]. We divided our method in three steps: (1) region selection and occlusion detection; (2) reading order arrangement; and (3) balloon generation and placement. Figure 4 illustrates the proposed method for word balloon placement. First, for each speaking character (C_1 and C_2), four candidate regions (C_iR_1, C_iR_2, C_iR_3, and C_iR_4) are generated around the character's face for the placement of word balloons. The region of each word balloon is selected according to weights associated with the available candidate regions based on their importance for the layout of the panel. We consider, for example, a balloon placed in front of a speaking character to be better than one placed on its back, so candidate regions in front of the speaking character have a higher weight associated with them. If a region overlaps an occupied candidate region of another character that speaks first, the region is marked as unavailable (C_2R_1). The same happens when the candidate region overlaps another character or interactive object. In the second step, the selected regions are expanded (EC_1R_2, EC_2R_3) to allow the arrangement of the word balloons according to the reading order of comics, where word balloons must be placed in a left-right and top-down order. Finally, in step 3, the word balloons are generated and placed in their corresponding regions.



(Step 1) Region Selection and Occlusion Detection

(Step 2) Reading Order Arrangement

(Step 3) Balloon Generation and Placement

Fig. 4. Balloon placement process.

After placing the characters, interactive objects and word balloons, the size of the image frame being generated is adjusted to match the actual size of its respective panel. Panels that don't include changes of the emotional state of the characters are framed so that all the characters, interactive objects and word balloons stay visible and centered in the panel. Panels that include modification of the emotional state of the characters are framed starting from the character's hips, which emphasizes the emotional expressions of the character's face.

7 User Interaction

The proposed method for user interaction is based on interactive objects. Through this device users are allowed to interact with the narrative by changing the content of the panels, and, consequently, altering the logical context of the story. Such interventions may force the characters to take other actions and change the course of the narrative.

Each interactive object includes a list of effects consisting of facts to be added or removed from the current state of the world when the object is activated. The story events generated by the planner are associated with a description of the current state of the world when the event occurred during the planning process. When an interactive object is activated by the user, the state of the world at that specific point of the narrative is modified according to the predefined effects of the activated object (*i.e.* facts are added or removed from the state).

To complement the world state modification operated by the user interaction, the system requests a new plot for the story planner, so as to create an alternative story that is consistent with the changes caused by the activation of the object. In this case, the modified state is sent to the planner as the initial state of the story. In the new plan, the previous events remain unaffected, whereas the effects of the user intervention are propagated to the next story events. Consequently, the comic panels are updated to reflect the new storyline.

In order to exemplify the user interaction process, let us consider the example illustrated in Figure 5. In this example, an antidote bottle is an interactive object that has the effect: ¬*at*(*antidote, hospital_room_1*), where the negation symbol (¬) indicates a fact to be removed from the world state when the object is activated. In this case, the effect removes the fact that the antidote exists in hospital room number 1. The current state of the world during the event where the antidote bottle is visible (panel 3 on Figure 5a) is described by the following facts: *healthy*(*anne*), *healthy*(*john*), *healthy*(*jimmy*), *wasinfected*(*emma*), ***at*(*antidote, hospital_room_1*)**, *at*(*jimmy*, *house*), *at*(*emma*, *house*), *at*(*anne*, *house*), *at*(*john*, *hospital_room_1*). If the user touches the antidote bottle (Figure 5b), the antidote will be dropped on the table, and the effects of the interactive object will be applied. In particular, the fact *at*(*antidote*, *hospital_room_1*) is removed from the world state of the event where the interaction occurred. A new story plot is then requested to the story planner, where, as explained before, the modified world state is now the initial state for the plan. As a result, a new plot describing what happened next is generated and the comic panels are updated according to the new storyline (Figure 5c).

(a) (b) (c)

Fig. 5. User interaction.

8 Application and Evaluation

The prototype application developed to test our system pertains to a zombie survival genre. It tells the story of a family that lives in a world dominated by a zombie plague. The main characters of the narrative are: the brave husband, Mr. John; his beautiful wife, Anne; and their children, Emma and Jimmy, who are always getting in trouble.

In the main storyline, one of the characters is attacked and infected by a zombie and the family tries to get an antidote to save the victim's life. The story takes place in three main locations: the family house, a dangerous park, and a hospital. Users can interact with the story through the following interactive objects: (1) house door - opening the door of the family house allows a zombie to enter in the house and possibly attack someone; (2) old trunk - opening an old trunk in the family house reveals a toy that can entertain the children; (3) zombie at the park - pushing a zombie in the park makes the zombie fall and die; (4) hospital main door - closing the main door of the hospital prevents the entrance/exit of the characters in/from the hospital through the main door, while opening the door allows their entrance/exit; (5) hospital back door - closing the main back door of the hospital prevents the entrance/exit of the characters in/from the hospital through the back door, while opening the door allows the entrance/exit; and (6) antidote bottle - pushing the antidote bottle makes it fall and spill the antidote. The interactive objects can have different effects in the story depending on which part of the narrative is occurring when they are activated.

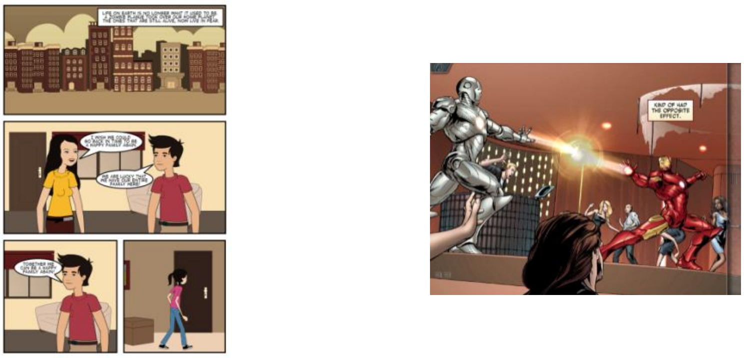
The prototype application is able to generate a considerable number of diversified stories. In the more conventional stories, Emma is attacked by a zombie in the park and then saved by her father who finds an antidote in the hospital; in stories with a not so happy ending, John does not find the antidote and decides to kill his daughter in order to protect the others; and in stories with an even darker outcome, John gets stuck in the hospital and cannot return home with the antidote, the daughter turns into a zombie, kills the rest of the family, and John commits suicide in the hospital.

8.1 User Evaluation

In order to evaluate our system, we have conducted a user evaluation with 18 high school students, 15 male and 3 female, aged 16-17. Six of them had previous experiences with interactive storytelling systems. Twelve of them play video games at least weekly. Sixteen of them like comic books, of which four read comics at least weekly.

We asked participants to use both our interactive storytelling system (S) and the mobile interactive comic book *The Avengers: Iron Man - Mark VII* [34] (M) for Android (Figure 6). We aimed to compare the proposed system with the "state of the art" of interactivity in comics. In order to avoid the focus on the appealing art quality of Marvel's system, we asked participants to concentrate their attention mainly on the application flow, user interaction, and the way the story is presented. And to reduce learning effects, half of the participants used S first, and the other half used M first. On average, each session of S lasted 12.3 minutes (σ=2.4) and included 5.8 interactions (σ=2.2). In M, each session lasted 7.5 minutes (σ=1.1).

After using each version, the participants filled out a questionnaire with 25 questions derived from the IRIS Evaluation Toolkit [27][28]. We evaluated the system usability, the correspondence of system capabilities with user expectations (user satisfaction), the interaction effectiveness and the users' experience (curiosity, flow and enjoyment). Each statement was followed by a seven-point Likert scale ranging from "strongly disagree" (-3) through "neutral" (0) to "strongly agree" (+3). After having interacted with both systems, the participants were interviewed about their experience.



(*a*) Our Interactive Comics System (*b*) The Avengers: Iron Man - Mark VII

Fig. 6. Visual comparison of the evaluated systems. Our system presents several panels on the screen (*a*), while Marvel’s system presents only one panel per screen (*b*).

Figure 7 summarizes the results of the questionnaires. Our interactive comics system clearly obtained better results in the evaluated topics (system usability, interaction effectiveness, user satisfaction, and user experience). *Marvel's The Avengers: Iron Man - Mark VII* obtained very low interaction effectiveness, mainly because it has a linear storyline and user interactions don’t have any real effect in the story plot. As far as the interviews are concerned, sixteen of the participants stated that they preferred our interactive comics system. They declared that the layout of our system is more similar to real comic books, and the possibility of changing the story is very attractive and innovative. According to them, Marvel’s system is more similar to a game than to a comic book. Following are some quotes from users regarding their experience while using our system:

“*The experience was amazing! It was really cool to see the panels and the story changing according to my actions.*” (P1)

“*The ability to interfere in the story was very exciting. I felt very curious and interested in finding out everything that my actions could do, and what else might happen in the story*.” (P2)

“*I really enjoyed the possibility of changing the story by touching the objects in the panels*.” (P3)

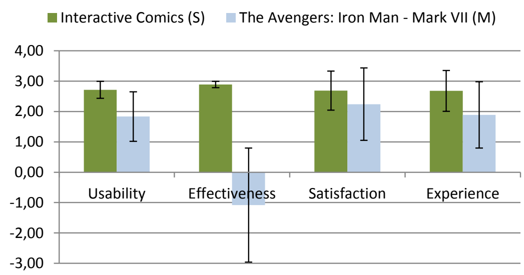


Fig. 7. Average and standard deviation of questionnaire topics for our *Interactive Comics* system and Marvel's *The Avengers: Iron Man - Mark VII* system.

9 Conclusion

In this paper we presented a comics-based interactive storytelling system capable of generating dynamic interactive narratives in the format of comic books. The system combines planning techniques for the dynamic creation of interactive plots with intelligent algorithms for the automatic generation of comics. We believe that non-branching interactive comics can expand the boundaries of the classical style of comic books towards a new form of digital storytelling.

Although the preliminary results of the user study are still inconclusive due to the small number of participants, their positive feedback suggests the effectiveness of the proposed approach and indicates a promising direction for future research.

For the sake of conciseness, in this paper we decided to leave out details covered in our previous works on how to automatically define the type of shot, type of panel transitions, and scene illumination depending on the characters' emotions and the comic artist's style [32][33]. Those adaptations would substantially improve the capacity of the system in dealing with other artistic aspects of the generated comics, while keeping the signature of the human artist.

References

1. Kurlander, D., Skelly, T., Salesin, D.: Comic Chat. In: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques, pp. 225-236 (1996)
2. Sumi, Y., Sakamoto, R., Nakao, K., Mase, K.: ComicDiary: Representing individual experiences in a comics style. In: Proceedings of the 4th International Conference on Ubiquitous Computing, LNCS 2498, Springer, pp. 16-32 (2002)
3. Shamir, A., Rubinstein, M., Levinboim, T.: Generating Comics from 3D Interactive Computer Graphics. IEEE Computer Graphics and Applications, 26 (3), pp. 53-61 (2006).
4. Shuda, T., Thawonmas, R.: Frame Selection for Automatic Comic Generation from Game Log. In: Proceedings of the 7th International Conference on Entertainment Computing (ICEC 2008), LNCS 5309, Springer, Heidelberg, pp. 179-184 (2008)
5. Chan, C.J., Thawonmas, R., Chen, K.T.: Automatic Storytelling in Comics: A Case Study on World of Warcraft. In: Proceedings of the 27th International Conference on Human factors in Computing Systems (CHI 2009), pp. 3589-3594 (2009)
6. Pizzi, D., Lugrin, J., Whittaker, A., Cavazza, M.: Automatic Generation of Game Level Solutions as Storyboards. IEEE Transactions on Computational Intelligence and AI in Games, 2 (3), pp. 149-161 (2010)
7. Alves, T., Mcmichael, A., Simões, A., Vala, M., Paiva, A., Aylett, R.: Comics2D: Describing and Creating Comics from Story-Based Applications with Autonomous Characters. In: Proceedings of the International Conference on Computer Animation and Social Agents, Belgium (2007)
8. Alves T., Simões A., Figueiredo R., Vala M., Paiva A., Aylett R.: So tell me what happened: Turning agent-based interactive drama into comics. In: Proceedings of the 7th International Conference on Autonomous Agents and Multiagent Systems, Portugal, pp. 1269-1272 (2008)
9. Aylett, R., Louchart, S., Dias, J., Paiva, A.: FearNot! An Experiment in Emergent Narrative. In: Proceedings of the 5th International Conference on Intelligent Virtual Agents, pp. 305-316 (2005)

10. Spierling, U., Braun, N., Iurgel, I., Grasbon, D.: Setting the scene: playing digital director in interactive storytelling and creation. Computers & Graphics, v. 26, n.1, pp. 31-44 (2002)
11. Cavazza, M., Charles, F., Mead, S.: Character-based interactive storytelling. IEEE Intelligent Systems, Special issue AI in Interactive Entertainment, vol. 17, (4), pp. 17-24 (2002)
12. Mateas, M.: Interactive Drama, Art, and Artificial Intelligence. Ph.D. Thesis. School of Computer Science, Carnegie Mellon University, Pittsburgh, USA (2002)
13. Bonet, B., Geffner, H.: Planning as Heuristic Search. Artificial Intelligence Special Issue on Heuristic Search, 129, n.1, pp. 5-33 (2001)
14. Pearl, J.: Heuristics: Intelligent Search Strategies for Computer Problem Solving, Addison-Wesley (1985).
15. Ghallab, M., Nau, D., Traverso, P.: Automated Planning: Theory and Practice. 1 ed. Amsterdam, Morgan Kaufmann Publishers (2004)
16. Witty, S.: Illustrative Text: Transformed Words and the Language of Comics. In: International Conference of Arts and Humanities, Honolulu, Hawaii, (2007).
17. McCloud, S.: Understanding Comics. Kitchen Sink Press. Northampton, MA (1993)
18. Eisner, W.: Comics & Sequential Art. Poorhouse Press. Tamarac, FL (1985)
19. Goodbrey, D. M.: PoCom-UK-001. Available at: http://e-merl.com/pocom.htm, 2003
20. Goodbrey, D. M.: The Archivist. Available at: http://e-merl.com/archivist/, 2010
21. McCloud, S.: Reinventing Comics: How Imagination and Technology Are Revolutionizing an Art Form. William Morrow Paperbacks (2000).
22. Campbell, S.: Nawlz – Interactive Comic. Available at: http://www.nawlz.com/hq/about/
23. Andrews, D., Baber, C., Efremov, S., Komarov, M.: Creating and Using Interactive Narratives: Reading and Writing Branching Comics. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2012), pp. 1703-1712 (2012)
24. Fikes, R., Nilsson, N.: STRIPS: A new approach to the application of theorem proving to problem solving. Artificial Intellligence, 2, pp. 189-208 (1971)
25. Chun, B., Ryu, D., Hwang, W., Cho, H.: An Automated Procedure for Word Balloon Placement in Cinema Comics. ISVC, 2, pp. 576-585 (2006)
26. Riedl, M.O., Young, R.M.: From Linear Story Generation to Branching Story Graphs. IEEE Computer Graphics and Applications, Vol. 26, No. 3 (2006).
27. Klimmt, C., Roth, C., Vermeulen, I., Vorderer, P.: The Empirical Assessment of The User Experience In Interactive Storytelling: Construct Validation Of Candidate Evaluation Measures. Technical Report, Integrating Research in Interactive Storytelling - IRIS (2010).
28. Roth, C., Vorderer, P. Klimmt, C.: The Motivational Appeal of Interactive Storytelling: Towards a Dimensional Model of the User Experience. In: International Conference on Interactive Digital Storytelling, LNCS vol. 5915, Springer, pp. 38-43 (2009)
29. O'Neil, D.: The DC Comics Guide to Writing Comics. Watson-Guptill, 1st Ed. (2001)
30. Shiga, J., Plotkin, A.: Meanwhile. Available at: http://zarfhome.com/meanwhile
31. Fenty, S., Houp, T., Taylor, L.: Webcomics: The Influence and Continuation of the Comix Revolution. ImageTexT: Interdisciplinary Comics Studies, Univ. of Florida, vol. 1 (2004).
32. Lima, E.S., Pozzer, C.T., Feijo, B., Ciarlini, A.E.M., Furtado, A.L.: Director of Photography and Music Director for Interactive Storytelling. In: IX Brazilian Symposium on Games and Digital Entertainment - SBGames 2010, Brazil. pp. 122-131 (2010)
33. Lima, E.S., Feijo, B., Furtado, A.L., Pozzer, C.T., Ciarlini, A.E.M.: Automatic Video Editing For Video-Based Interactive Storytelling. In: Proceedings of the 2012 IEEE International Conference on Multimedia and Expo (ICME 2012), Melbourne, pp. 806-811 (2012)
34. Loudcrow Interactive. Marvel's The Avengers: Iron Man – Mark VII, Available at: http://loudcrow.com/marvels-the-avengers-iron-man-mark-vii