

Event Relations in Plan-Based Plot Composition

Angelo E. M. Ciarlini¹ Simone D. J. Barbosa²
¹UNIRIO, Depto. de Informática Aplicada, Brasil

Marco A. Casanova² Antonio L. Furtado²
²PUC-Rio, Depto. de Informática, Brasil

Abstract

The process of plot composition in the context of interactive storytelling is considered under a fourfold perspective, in view of syntagmatic, paradigmatic, antithetic and meronymic relations between the constituent events. These relations are shown to be associated with the four major tropes of semiotic research. A conceptual model and set of facilities for interactive plot composition and adaptation dealing with the four relations is described. To accommodate antithetic relations, corresponding to the irony trope, our plan-based approach leaves room for the unplanned. A simple storyboarding prototype tool has been implemented to conduct experiments.

Keywords: Interactive Storytelling, Plots, Planning, Narratology, Tropes.

Authors' contact:

angelo.ciarlini@uniriotec.br
 {simone,casanova,furtado}@inf.puc-rio.br

1. Introduction

The role of storytelling in games has long been the subject of lively debates [Wardrip-Fruin & Harrigan]. Although some believe that story and game are in direct opposition [Costikyan], most agree that successful narrative in games is possible, and a few argue for the importance of story creation as part of gameplay [Wallis]. However, a different sort of narrative is required: it must be non-linear and play-centric, that is, it must revolve around the player's experience [Pearce]. The player is no longer a mere consumer of the narrative, but both a consumer and a (co-) producer of the plot. The game designer typically selects a genre. In game playing, interactive storytelling emerges, but care must be taken to ensure that the basic rules of the genre, as well as corresponding tropes and narrative structures, are understood by the co-authors of the story [Wallis].

A few computational systems and approaches have been proposed to support interactive storytelling. Some of them focus on the interaction among characters [Cavazza et al.], whereas others focus on plot structure and coherence [Grasbon & Braun], and a few others attempt to combine both [Mateas & Stern]. What kind of system would be suitable for assisting users in creating stories within games or other interactive storytelling contexts? *Planning algorithms* have proven to be a useful alternative to help create narratives by exploring different chains of events to achieve the characters' or the storytellers' goals [Ciarlini et al.; Riedl & Young]. In game playing, planning algorithms make it practical to create non-linear narratives that are both coherent and diverse, by allowing players to proceed in different courses of actions with varying results, and yet respecting the game structure, rules and constraints.

To support the production of stories, we have drawn on what semiotic research has singled out as the *four major tropes* [Burke], namely: metaphor, metonymy,

synecdoche, and irony. By offering mechanisms derived from these tropes, we intend both to augment the expressiveness of narrative models and to provide better support to authors who are less familiar with or confident in creating and telling stories.

In this paper, we associate those tropes with four types of relations between narrative events: *syntagmatic*, *paradigmatic*, *meronymic* and *antithetic*. They play a basic role in an interactive plan generating system that creates plots within a predefined genre.

Narratology studies distinguish three levels in literary composition: *fabula*, *story* and *text* [Bal]. In the present work, we stay at the *fabula* level, where the characters acting in the narrative are introduced, as well as the narrative plot, consisting of a partially-ordered set of events. We focus on plots whose constituent events happen as a consequence of a predefined repertoire of actions, which we shall call *operations*, deliberately performed by the characters. Plot composition will be treated here as a plan generation process, and hence the terms *plot* and *plan* will be used interchangeably. Yet, since narratives are often more attractive when unplanned shifts can occur, the user shall retain the power to issue certain *directives* when interventions are needed or desired.

Starting from such considerations, this paper proposes a fourfold way to characterize plot composition at the *fabula* level. Section 2 describes the relations between events in correspondence with the four major tropes. Section 3 outlines how we model an intended genre, to whose conventions the plots must conform. Section 4 sketches, over a simple example, the main features of our plan-based prototype tool. Concluding remarks are presented in section 5.

2. From Tropes to Event Relations

It has been suggested that the four major rhetorical tropes provide models for remarkably comprehensive analyses in different areas [Burke; Chandler; White]. They all involve *relations between pairs of words*, thanks to which, given two related words w_1 and w_2 , a person can meaningfully use w_1 to refer to w_2 .

They are not defined in a uniform way by linguists, there being much disagreement, especially on the distinction between metonymy and synecdoche. A useful discussion is found in [Chandler], where many practical applications of Burke's four tropes theory are surveyed.

Metaphor [Lakoff & Johnson; Ortony] and synecdoche [Chandler] have to do with hierarchical structures such as those represented in ontologies [Breitman, Casanova & Truszkowski]. If one concept C_1 can be metaphorically used to denote another concept C_2 , the two concepts are said to be similar or analogous, and are placed under a more general concept \hat{C} that subsumes both of them. C_1 and C_2 would be represented in the network with **is-a** links connecting them to \hat{C} . Also, one could add an **is-like** link from C_1 to C_2 [Breitman et al.]. Clearly, metaphor is a displacement along the verbal

paradigmatic axis [Saussure], from which we took the suggestion of a *paradigmatic relation* between events.

In synecdoche, concept C_1 is used to denote concept C_2 , if C_1 is a part of C_2 (which calls for another link, C_1 **part-of** C_2); the converse substitution, from whole to part, is also usual in common parlance. The corresponding association between events is called *meronymic relation* in the present paper.

According to [Chandler], metonyms are based on various indexical relationships between concepts, notably the substitution of effect for cause, and convey an idea of contiguity. Borrowing again from [Saussure], we require the presence of *syntagmatic relations* between events, to justify their being meaningfully placed in sequence.

Irony is the most intriguing of the four tropes. In verbal communication, it reflects the opposite of the thoughts or feelings of the speaker or writer (as when you say 'I love it' when you hate it) or the opposite of the truth about external reality (as in 'there's a crowd here' when it is deserted). It also takes the form of substitution by *dissimilarity* or *disjunction*. Variations such as understatement and overstatement can also be regarded as ironic. At some point, exaggeration may slide into irony [Chandler]. Disclosing paradoxes and hidden agendas in literary texts, in sharp contrast between the declared intentions and the real ones, is another source of irony, constituting a trend in critical studies known as *deconstruction* [Culler].

Not only mental attitudes, feelings and statements can be ironic – actions can also be ironic, but always in an unplanned, non-deliberate fashion. Irony is in fact a characteristic of certain intrigue situations that are often referred to as *dramatic irony* [Booth].

Consequently any kind of irony induces an *antithetic relation* between events that look, in principle, incompatible with each other, given their dependence on contexts characterized by radically opposite properties. Mediating two such events, the until then well-behaved world must suffer a disruptive shift, whereby the truth value of certain facts or beliefs is inverted, or certain properties move from one extreme to the other within the ascribed value range (e.g. from helplessly weak to heroically strong).

To illustrate the event relations derived from the major tropes, we shall employ a simple example to be referenced along the paper. Consider four types of events, all having one woman and two men as protagonists: *abduction*, *elopement*, *rescue*, and *capture*. As demonstrated in folktale studies [Propp], many plots mainly consist of an act of villainy, i.e. of a violent action that breaks the initially stable and peaceful state of affairs, followed ultimately by an action of retaliation, which may or may not lead to a happy outcome.

Propp distinguished seven character roles (*dramatis personae*) according to the events assigned to each one's initiative: hero, villain, victim, dispatcher, donor, helper, false hero. Curiously, in literary texts involving the four events above, this distribution is not unique: we called the violent initial act “villainy”, but the perpetrator of abduction, and more often of elopement, can be the hero of the narrative, and in such cases the woman's original guardian (husband, father) is regarded as the villain.

2.1 Syntagmatic Relations

To declare that it is legitimate to continue a plot containing abduction by placing rescue next to it, we say that these two events are connected by a *syntagmatic relation*. More precisely, we can define the semantics of the two

events in a way that indicates that the occurrence of the first leaves the world in a state wherein the occurrence of the second is coherent. Similarly, a plot involving elopement followed by capture looks natural, and hence these two events are likewise related.

The syntagmatic relation between events induces a weak form of causality or enablement, which justifies their *sequential ordering* inside the plot.

2.2 Paradigmatic Relations

The events of abduction and elopement can be seen as *alternative* ways to accomplish a similar kind of villainy. Both achieve approximately – though not quite – the same effect: one man takes away a woman from where she is and starts to live in her company at some other place. There are differences, of course, since the woman's behaviour is usually said to be coerced in the case of abduction, but quite voluntary in the case of elopement. In fact, it is usual to assume that a sentence such as “Helen elopes with Paris”, implies that Helen had fallen in love with Paris.

To express that abduction and elopement play a similar function, we say that there is a *paradigmatic relation* between the two events. Likewise, this type of relation is perceived to hold between the events of rescue and capture, which are alternative forms of retaliation. And, again, there is a difference between the woman's assumed attitude, associated as before with her feelings. An abducted woman expects to be rescued from the villain's captivity by the man she loves. On the contrary, she will only return through forceful capture if she freely eloped with the seducer.

As the present example suggests, the syntagmatic and the paradigmatic axes identified by Saussure are really *not* orthogonal in that the two relations cannot be considered independently when composing a plot. Thus, in principle, the two pairs enumerated in the previous section (abduction-rescue and elopement-capture) are the only normal combinations, the former illustrated by the Sanskrit *Ramayana* [Valmiki] and the similarly structured Arthurian romance of *Lancelot* [Chrétien; Furtado & Veloso], and the latter by the Irish *Story of Deirdre* [McGarry]. Yet the next section shows that such limitations can, and even should, be waived occasionally.

2.3 Antithetic relations

While normal plots, whose outcome is fully determined, can be composed exclusively on the basis of the two preceding relations, the possibility to introduce unexpected turns is often desirable in order to make the plots more attractive – and this requires the construct that we chose to call *antithetic relation*. A context where a woman suffers abduction by a ravisher whom she does not love would seem incompatible with a capture event, since there should be no need to employ force to bring back the victim. So, in this sense, abduction and capture are in antithetic relation.

The mythical *Rape of the Sabines* shows what can happen as a consequence of a drastic reversal of the circumstances. King Romulus is facing a problem at the newly founded city of Rome: the population is entirely male at first. To remedy the lack, he leads his men to break into the dwellings of the Sabines and abduct their women. Sometime afterwards the Sabine warriors march against the Romans, but the women have no wish to be taken back, leaving to their countrymen no option except

their capture. King Romulus's men had lawfully married them and made them bear children. A Roman chronicle [Titus Livius] reports the radical change in the women's feelings, and tells how the seemingly inevitable confrontation ended with the reconciliation of the two parties.

In contrast, modern history provides some distinctly regrettable examples of abduction actually followed by capture, categorized by psychiatrist Nils Bejerot as the *Stockholm syndrome*. One case in point is the abduction by a group of terrorists of the daughter of a millionaire, who ended up joining her tormentors in the practice of crimes, and was captured by the police in San Francisco [Hearst & Moscow].

The occurrence of elopement followed by rescue provides a much stronger case of antithetic relation. Indeed, elopement only makes sense if the victim loves the seducer, whereas, for this very motive, she would resist to any attempt to rescue her, leaving forceful capture as the only viable alternative. Even so the legendary story of *Helen of Troy*, in spite of various discordant interpretations, seems to offer a counter-example. Married to king Menelaus of Sparta, Helen fled to Troy in the company of Paris, out of her free will according to a number of versions (e.g. the *Heroides* [Ovid]). But, after their escapade to Troy where they married, her love feelings started to wane while the Trojan War followed its bloody course and she kept recalling the far manlier Menelaus. The *Iliad* [Homer] signals repeatedly this critical change of sentiment. At the end her recovery turned from capture into rescue, as registered in the *Aeneid* [Virgil]. Paris was dead, and she had been delivered to Paris's brother Deiphobus. When the Greeks came out of the wooden horse and stormed the Trojan palaces, Helen herself made sure that Menelaus should win – and know that she was helping him in atonement for her previous misconduct. The shadow of Deiphobus tells the episode to Aeneas; and what better example of irony could we find than his calling Helen “this peerless wife”?

One more example appears in the story of *Tristan and Isolde*, in several versions [Marchello-Nizia]. The knight had eloped with the queen; they were living in harsh conditions in a forest. The dramatic change of their love feelings, which allowed Isolde's rescue by king Mark to be achieved through a simple invitation, with no need to fight, had a very curious cause – the timely expiry date of the love potion they had drunk before, when sailing from Ireland to Cornwall [Bérroul].

Generally speaking, if some *binary opposition* – the “to love or not to love” dilemma, in the present case – is allowed to be manipulated via some agency external to the predefined events, then one can have plots that no longer look conventional. A sort of discontinuity is produced by such radical shifts in the context. Intervening between abduction and capture, or between elopement and rescue, a sudden change of feelings can give rise to these surprising sequences. Also, both in fiction and in reality, things not always proceed according to planned events. Natural phenomena and disasters, the mere passage of time, the intervention of agents empowered to change the rules, supernatural or magic manifestations, etc., cannot be discounted.

Specifically for the tragedy genre, the *Poetics* [Aristotle] distinguishes between simple and complex plots, characterizing the latter by the occurrence of *recognition* (ἀναγνориσις) and *reversal* (περιπέτεια). Differently

from reversal, recognition does not imply that the world itself has changed, but rather the *beliefs* of one or more characters about the actual facts. Because of a change of beliefs, a reason to be added to those enumerated in the previous paragraph, a reversal in the course of actions can take place, usually in a direction totally opposite to what was going on so far. Yet another possible external cause of both recognition and reversal in the tragic scene was the intervention of a god, who was lowered onto the stage using a crane – known, accordingly, as *deus ex machina*.

Aristotle's remarks are clearly relevant to the present discussion of plots in general. Following his lead, we shall admit state changes outside the regular regime of predefined events by allowing the user – literally acting *ex machina* (via the computer...) – to impose variations to the context (both in terms of *facts* and of *beliefs*), and thereby deviate the action from its predicted path.

This extreme device will be necessary to allow the elopement-rescue sequence. We decided, however, not to make it indispensable for abduction-capture, in order to have a chance to present a good example of *erroneous* beliefs, contradicting the actual facts. Criminal records everywhere are full of simulated abduction pacts for drawing a ransom from a deluded family. Conversely, a man can unnecessarily decide that capture is the only way to bring back a woman, if he mistakenly believes her to love the ravisher.

Figure 1 shows the relations thus far discussed.

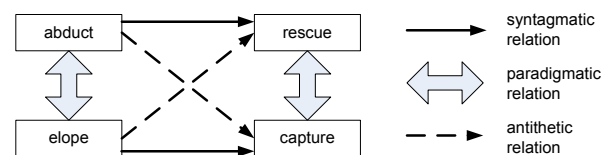


Figure 1: Syntagmatic, paradigmatic, and antithetic relations.

2.4 Meronymic relations

Meronymy is a word of Greek origin, used in linguistics to refer to the decomposition of a whole into its constituent parts. Forming an adjective from this noun, we shall call *meronymic relations* those that hold between an event and a lower-level set of events, with whose help it is possible to provide a more detailed account of the action on hand.

Thus, we could describe the abduction of a woman called Sita by a man called Ravana (characters taken from the *Ramayana* [Valmiki]) as: “Ravana rides from Lanka to forest. Ravana seizes Sita. Ravana carries Sita to Lanka.” And her rescue by Rama could take the form: “Rama rides from palace to Lanka. Rama defeats Ravana. Rama entertreats Sita. Rama carries Sita to palace.” But notice that such decompositions are not fixed, since the lower-level events are selected as required by the current state. For instance, with respect to the rescue event, the hero may already be present at the ravisher's dwelling, or perhaps the victim is not held in captivity, respectively obviating the need for the voyage or for fighting the enemy (Figure 2).

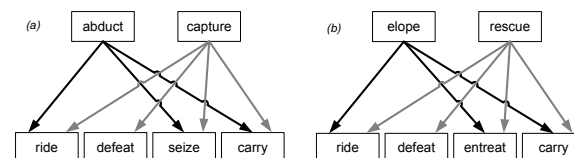


Figure 2: Meronymic relations: (a) the forceful actions and (b) the gentle actions.

Detailing is most useful to pass from a somewhat abstract view of the plot to one, at a more concrete physical level, that is amenable (possibly after further decomposition stages) to the production of a computer graphics animation [Ciarlini et al.]. Mixed plots, combining events of different levels, do also make sense, satisfying the option to represent some events more compactly while showing the others in detail.

The intuitive notions behind figures 1 and 2 are partly derivable from a context-sensitive grammar:

```
PLOT ::= VILLAINY•RETALIATION
VILLAINY ::= ABDUCT | ELOPE
RETALIATION ::= RESCUE | CAPTURE
ABDUCT•RESCUE ::= abduct, rescue
ELOPE•CAPTURE ::= elope, capture
ABDUCT•CAPTURE ::= *(abduct, capture)belief
ELOPE•RESCUE ::= *(elope, rescue)fact
ABDUCT•RESCUE ::= ABDUCT2•RESCUE2
ELOPE•CAPTURE ::= ELOPE2•CAPTURE2
ABDUCT•CAPTURE ::= *(ABDUCT2•CAPTURE2)belief
ELOPE•RESCUE ::= *(ELOPE2•RESCUE2)fact
ABDUCT2 ::= ride, seize, carry
RESCUE2 ::= ride, defeat, entreat, carry
ELOPE2 ::= ride, entreat, carry
CAPTURE2 ::= ride, defeat, seize, carry
```

3. A Plan-based Modelling Approach

To model a chosen genre, to which the plots to be composed should belong, we must specify at least (to be the object of section 3.1):

- a. what can exist at some state of the underlying mini-world,
- b. how states can be changed, and
- c. the factors driving the characters to act.

In our model, we equate the notion of event with the state change resulting from the execution of a predefined operation. Being defined in terms of their pre-conditions and post-conditions, operations can be readily chained together by a *plan-generating algorithm* [Ciarlini et al.; Barros & Musse] in order to achieve a given goal of some character. As a consequence, it becomes natural to equate plots (sequences of events) with plans (sequences of operations able to bring about the events). Also, to confer a degree of autonomy [Riedl & Young] to the characters performing the operations, it is convenient to make their goals emerge from appropriately motivating situations.

Viewing plots as plans suggests an obvious plot composition strategy, having a plan-generator as its main engine. This and the fact that our conceptual model is expressed in Prolog make the genre specification executable. In sections 3.2, 3.3 and 3.5, we will argue that, duly complemented by auxiliary routines, the planning strategy deals effectively with narrative plots in view of three out of the four event relations. To accommodate antithetic relations, however, it will be necessary to leave room for the *unplanned*, as proposed in section 3.4, leading to plots that may to a limited extent break the conventions of the adopted genre.

3.1 Conceptual schemas

We start with a conceptual design method involving three schemas – static, dynamic and behavioural – which has been developed for modelling literary genres encompassing narratives with a high degree of regularity, such as fairy tales, and application domains of business information systems, such as banking, which are obviously constrained by providing a basically inflexible set of operations and, generally, by following strict and

explicitly formulated rules [Furtado et al. 2008]. For brevity, the detailed logic programming notation is omitted; the full specification is shown in Appendix A of our technical report¹.

The *static schema* specifies, in terms of the *Entity-Relationship* model [Batini et al.], the entity and relationship classes and their attributes. In our simple example, `character` and `place` are entities. The attributes of characters are `name`, which serves as identifier, and `gender`. Places have only one identifying attribute, `pname`. Characters are pair-wise related by relationships `loves`, `held_by` and `consents_with`. The last two can only hold between a female and a male character; thus `held_by(Sita,Ravana)` is a *fact* meaning that Sita is forcefully constrained by Ravana, whereas `consents_with(Sita,Ravana)` would indicate that Sita has voluntarily accepted Ravana's proposals. Two relationships associate characters with places: `home` and `current_place`. A state of the world consists of all facts about the existing entity instances and their properties holding at some instant.

The *dynamic schema* defines a fixed repertoire of operations for consistently performing state changes. The *STRIPS* [Fikes & Nilsson] model is used. Each operation is defined in terms of pre-conditions, which consist of conjunctions of positive and/or negative literals, and any number of post-conditions, consisting of facts to be asserted or retracted as the effect of executing the operation. Instances of facts such as `home` and `gender`, are fixed, not being affected by any operation. Of special interest are the *user-controlled* facts which, although also immune to operations, can be manipulated through arbitrary *directives* (cf. section 3.4). In our example, `loves` is user-controlled.

Again for the present example, we have provided operations at two levels. The four main events are performed by level-1 operations: `abduct`, `elope`, `rescue` and `capture`. Operations at level-2 are actions of smaller granularity, in terms of which the level-1 operations can be detailed: `ride`, `entreat`, `seize`, `defeat`, and `carry`.

Our provisional version of the *behavioural schema* consists of goal-inference (a.k.a. situation-objective) rules, belief rules, and emotional condition rules.

For the example, three goal-inference rules are supplied. The first one refers to the ravisher. In words, in a situation where the princess is not at her home and the hero is not in her company – and hence she is unprotected – the ravisher will want to do whatever is adequate to bring her to his home. The other goal-inference rules refer to the hero, in two different situations having in common the fact that the ravisher has the woman in his home: either the hero believes that she does not love the other man, or he believes that she does. In both situations, he will want to bring her back, freely in the first case and constrained in the second.

Informally speaking, beliefs correspond to the partial view, not necessarily correct, that a character currently forms about the factual context (for a formal characterization, cf. the BDI model [Cohen & Levesque; Rao & Georgeff]). The belief rules that we formulated for our example look rational, but notice that they are treated as

¹ ftp://ftp.inf.puc-rio.br/pub/docs/techreports/08_30_barbosa.pdf

defaults, which can be overruled as will be described in section 3.4. A man (the hero or the ravisher) believes that the woman does *not* love his rival if the latter has her confined, but if she has ever been observed in his company and in no occasion (state) was physically constrained, the conclusion will be that she is consenting (an attitude seemingly too subjective to be ascertained directly in a real context).

The emotional condition rules refer to the three characters. A man (or woman) is happy if currently in the company of his (or her) beloved, and bored otherwise. A special condition applies to the woman: she will be *absolutely* happy if, in addition to the first motive for contentment, she has never been constrained by any of the two adversaries.

3.2 Coherent sequences

Moving along the *syntagmatic* axis is primarily the task of the plan-generator, as it composes a coherent plot by aligning events in view of the pre- and post-conditions of the appropriate predefined operations.

For plot composition, it is convenient to proceed in a step-wise fashion, starting from a given initial state. At each state, the goal-inference rules are used to induce opportunistic short term goals from which successive plot sequences will originate.

In an interactive environment, at any step, the user, henceforward called the *Author*, should be allowed to intervene, reducing thereby the characters' autonomy, but relying on the plan-generator to enforce consistency within the genre. To this purpose, the Author may indicate a goal, to be tried by the plan-generator, or even a specific operation, which the plan-generator may or may not find applicable.

A more complex request is to indicate a sparse list of operations, to be filled-up until a valid plot sequence containing all operations in the list, possibly interspersed with others, is formed. The Author may optionally also indicate the desired goal, which would otherwise be assumed to coincide with the effects of the last operation in the list.

After the step-wise process terminates, it should still be possible to perform various kinds of adaptation. Those that have to do with the syntagmatic relations include adding or deleting operations and changing the sequence, if the partial order requirements imposed by the interplay of pre- and post-conditions permit. For instance, consider plot P below:

```
P = start => ride(Ravana, Lanka, forest) => entreat(Ravana, Sita)
=> seize(Ravana, Sita) => carry(Ravana,Sita,Lanka)
```

which can be re-ordered, to meet the Author's preferences, to produce:

```
Ps = start => ride(Ravana, Lanka, forest) => entreat(Ravana,
Sita) => carry(Ravana,Sita,Lanka) => seize(Ravana, Sita)
```

Curiously, both the original plan P and the reordered plan Ps suggest stories that may well happen in reality or fiction. In P, a voluntary elopement is disguised as an abduction, whereas in Ps elopement is cruelly followed by the woman's confinement.

Also, a plot can be extended with more operations if the Author supplies an additional goal in an attempt to provide a continuation.

3.3 Alternative choices

Moving along the *paradigmatic* axis gives ampler opportunity to obtain different plots than simply changing the sequence of events within the partial order requirements.

Alternatives may result, first of all, from starting from a different initial state, so that different goal-inference rules may be triggered. Notice also that more than one such rule may be ready for activation. In any case, the standard plan-generator's ability to *backtrack* is an expedient mechanism to engender alternative plots.

To resort to violence, as in abduction or capture, can be certainly regarded as excessive and unnecessary when the patient of the action loves the agent, even though our specification does not invalidate their occurrence. Accordingly, if the goal-inference rules are in control and the context is not tampered with (but see section 3.4), they will not figure in any generated plot. And yet the Author can have them as valid alternatives, simply by using the option to directly indicate a goal to the plan-generator. Such goal can be relatively non-specific, such as `current_place(Sita,palace)`, or else more restrictive, such as `(current_place(Sita,palace), held_by(Sita,Rama))` – in which case only the forceful capture event will result.

At the adaptation phase, the ability to replace one or more operations is a way to produce alternatives. One must bear in mind that a replacement may require another, if the Author is concerned with preserving consistency; so, replacing `abduct` by `elope` normally implies the replacement of `rescue` by `capture`.

A particularly convenient way to deal with entire plots, rather than with individual operations, is to take advantage of the similarity or analogy among situations, inherent in the notion of paradigms. Previously existing plots, no matter if composed manually or automatically, can be converted into *plot patterns* to be kept in a Library of Typical Plots [Furtado & Ciarlini 2001]. Plot patterns can then be *reused* to originate new plots, essentially by instantiating their variables in view of a new situation.

3.4 Shifts along the way

Until this point we restricted ourselves to planned and hence well-behaved plots. It is time now to introduce a measure of transgression, disrupting the context in order to obtain plots with events in *antithetic* relation.

The Author, as *deus ex machina*, can interfere with the plan generation discipline by issuing two kinds of *directives*, which can be applied both during composition and adaptation. One directive is `make_believe`, arbitrarily assigning a belief B to a character C, which overrules any previous belief on the same facts, either specified through the belief rules of the behavioural schema or stated by a previous application of the `make_believe` directive itself. If Sita was violently abducted by Ravana, Rama will believe (as a consequence of a belief rule) that she does not love the villain, and therefore that she will gladly consent to be rescued. However, the Author is allowed to induce Rama to falsely believe the contrary, which activates a goal-inference rule leading to a forceful `capture` event.

Another directive is `vary`, which manipulates user-controlled facts, instead of mere beliefs. In our example, the only facts declared to be user-controlled are the instances of the `loves` relationship, whose Boolean value will be inverted if the directive is applied. Sita can be eloped if she currently loves Ravana, and then be willingly rescued by Rama if between these two events the Author issues the directive so as to change her feelings. But `vary` does not have to be explicitly called for. A helpful feature in the course of plan-generation can detect

failures involving user-controlled facts, in which case the Author is asked whether or not the context should be tampered with accordingly.

In other example mini-worlds, one might have different kinds of user-controlled properties, e.g. with numerical values inside a range, such as degree of strength, which the *vary* directive could change in some radical proportion. Such representation is also appropriate for emotions in general, including love itself, enabling finely graded nuances of expression, obviously unattainable with simple two-valued Boolean alternatives.

We began to investigate another line, in an attempt to offer clues to an Author intent on finding ways to, at a later stage, replace the external *deus ex machina* directives by some internal narrative device with a flavour of irony, almost crossing the borderline of plausibility. Folktales, myths, and popular culture have pooled together through time rich repertoires of *motifs* [Aarne & Thompson], often containing ingenious solutions to dilemmas arising from antithetic situations.

Authors have always felt free to borrow from all kinds of sources, and one can easily discover occurrences of certain motifs in the literature of different countries, modified as required by cultural differences. For our example, we found three convenient motifs:

- a. life token: an object whose aspect changes if the owner is in distress,
- b. love potion: stimulates romantic/erotic feelings,
- c. ordeal: to vindicate a discredited or accused person.

where (a) (indexed as E761 in [Aarne & Thompson]) allows to do without the unrealistic assumption that characters are omniscient, e.g. explaining how Rama learned that Sita suffered abduction in the forest, (b) provides an excuse for sudden variations in amorous attachments, and (c) serves to restore the man's belief in his beloved's faithfulness. Curiously, both (b) and (c) occur in the *Tristan* romance, wherein the ordeal takes the especially ironic form of an ambiguous oath [Béroul], while in the *Ramayana* Sita has to walk through the fire [Valmiki]. In our example, we treat these motifs as black boxes, merely associating to their names a *<situation, goal>* specification. Thus, if the Author wants to insert motifs (simply through the mention of their names) at the positions in a generated plot where the respective situation holds, this can be asked for at the adaptation phase.

Such insertions are therefore to be regarded as provisional annotations only, which the Author should later have to unravel by mapping the events in the motifs into analogous events congenial to the genre adopted in the plot. The mappings should preserve the *<situation, goal>* of the motif and might require the definition of additional operations, such as communicative acts for instance. The persistence of motifs is a remarkable phenomenon, with relatively modern versions: microchip implants for (a), aphrodisiac drugs like the LSD hallucinogen for (b), and lie detectors and truth serums for (c), all of so dubious or controversial value as their primitive counterparts, but equally acceptable to the general public.

3.5 Down to details

As stated before, between level-1 and level-2 operations there may be *meronymic* relations. Creating plots in hierarchic fashion is a most common practice, starting with a broad view of the events, which in the case of our example corresponds to the level-1 operators. At later stages, one would gradually decompose each event into

finer grain actions, possibly along more than just two levels, to the point of coordinated physical movements, as required for displaying animated scenes [Ciarlini et al.].

When composing a plot, the plan-generator is free to mix operations of the two levels, a reasonable default option considering that the Author may wish to treat some events more succinctly than others. But the Author may, on the contrary, settle for a uniform style by indicating that only one of the two levels will be used. This choice can be altered at any time, in composition or adaptation.

Once a plot is composed, it can be adapted either by detailing or summarizing its constituent operations. Detailing each level-1 operation O_P in a plot into level-2 operations is treated as one more plan generation task, taking as *situation* the instantiated pre-conditions of O_P , and as *goal* the effects of O_P , and using exclusively the operations in the level-2 repertoire. More than one decomposition may be possible, depending on the initial state and on the changes effected by the preceding operations.

The inverse of detailing, summarizing, is also useful. We are currently restricted to a rather limited version, which only works if the detailed plan is divisible into subsequences that can be exactly subsumed by level-1 operations. This means that the process fails if other extraneous operations intervene. In other words, $summarize(P1, P2)$ succeeds if and only if $detail(P2, P1)$ also does.

Figures 2(a) and 2(b) are suggestive in that they illustrate a curious symmetry in how they map the example level-1 operations into level-2 operations. The decompositions in the two figures are the same, except for the substitution of *entreat* for *seize*. This is not surprising, since a similar decomposition comes as a consequence of the paradigmatic relation between the two villainy and the two retaliation events. Notice too that, in both figures, the event corresponding to villainy only differs from the retaliation event by the possible presence of *defeat* – reflecting our observation, after surveying a number of traditional narratives, that the villain almost always resorts to some trick, avoiding a confrontation that often (though not necessarily) occurs as part of retaliation.

The decompositions suggested by the two figures are typical but not unique, since the correspondence induced by the meronymic relations is not rigidly determined, i.e. it is, so to speak, context-sensitive, depending on the current state. For instance, *abduct* can be expressed by *seize* followed by *carry* if both the victim and the ravisher are currently at the same place, but will need a preliminary *ride* if the former is in the forest and the latter still in his home.

All this suggests that it may be difficult to interpret what is happening by looking at a sequence of level-2 operations without examining the context. In this regard, the ability to fill-up (cf. section 3.2) a sparse list of observed level-2 operations and then performing summarization, identifying what level-1 operation is taking place at some point, constitutes a not so trivial form of *plan-recognition* [Kautz]. Plan generation is more directly relevant to the composition and adaptation of plots than the recognition of plans and objectives. But the latter task is an asset in interactive plan-supported game-playing environments, since each player might employ it as an aid to discover what the opponents are trying to do.

4. A Prototype Implementation

A very simple prototype, **PlotBoard**, was designed to experiment with the notions discussed here. Dealing with *storyboarding* [Truong et al.] – exclusively at the fabula level – it serves to compose plots interactively with the help of an extended version of the early *Warplan* algorithm [Warren]. Written in SWI-Prolog², it interfaces with Java to show events in image format.

4.1 Some Features of the Plan-generator

The plan generator follows a backward chaining strategy. For a fact F (or $\text{not } F$) that is part of a given goal, it checks whether it is already true (or false) at the current state. If it is not, it looks for an operation Op declared to add (or delete) the fact as part of its effects. Having found such operation, it then checks whether the pre-condition Pr of Op currently holds – if not, it tries, recursively, to satisfy Pr . Moreover, the plan generator must consider the so-called frame problem [Lloyd], by establishing (in second-order logic notation) that the facts holding just before Op is executed stay valid unless explicitly declared to be altered as part of the effects of Op .

Like goals, pre-conditions are denoted by conjunctions of literals and arbitrary logical expressions. We distinguish, and treat differently, three cases for the involved positive or negative facts:

- a. facts which, in case of failure, should be treated as goals to be tried recursively by the plan generator;
- b. facts to be tested immediately before the execution of the operation, but which will not be treated as goals in case of failure: if they fail the operation simply cannot be applied;
- c. facts that are not declared as added or deleted by any of the predefined operations.

Note that the general format of a pre-condition clause is $\text{precond}(Op, Pr) :- B$. In cases (a) and (b), a fact F (or $\text{not } F$) must figure in Pr , with the distinction that the barred notation $/F$ (or $/(not F)$) will be used in case (b). Case (c) is handled in a particularly efficient way. Since it refers to facts that are invariant with respect to the operations, such facts are included in the body B of the clause, being simply tested against the current state when the clause is selected.

An example is the precondition clause of operation $\text{seize}(M, W)$, where M is the agent and W the patient of the action. Clearly the two characters should be together at the same place, and, accordingly, the Pr argument shows two terms containing the same variable P to express this requirement, but the term for W is barred: $/\text{current_place}(W, P)$, which does not happen in M 's case. The difference has an intuitive justification: the prospective agent has to go to the place where the patient is, but the latter will just happen to be there for some other reason.

The proper treatment of (a) and (b) is somewhat tricky. Suppose the pre-condition Pr of operation Op is tested at a state S_1 . If it fails, the terms belonging to case (a) will cause a recursive call whereby one or more additional operations will be inserted so as to move from S_1 to a state S_2 where Op itself can be included. It is only at S_2 , not at S_1 , that the barred terms in case (b) ought to be tested, and so the test must be *delayed* until the return

from the recursive call, when the plan sequence reaching S_2 will be fully instantiated.

Operations can admit more than one precondition clause, so as to cope with different circumstances. This happens with the $\text{carry}(M, W, P_2)$ operation, whereby W will either freely consent to be transported to P_2 by M , or will have to be forcefully held by him.

With respect to the added and deleted clauses declaring effects of operations, the plan generator also employs a barred notation, to distinguish between two cases: (a) primary effects, and (b) secondary unessential effects. In case (a), if any fact F to be added by Op already holds, or already does not hold if it should be deleted, then Op is considered *non-productive* and fails to be included in the plan. In contrast, in case (b), such lack of effect would be admitted and cause no failure.

As an example, consider the clause of operation $\text{capture}(M_1, W)$ that declares as deleted the fact $\text{held_by}(W, M_2)$, as a result of M_1 's action to take away W from M_2 . Notice that the fact may or may not hold prior to capture; it will hold if W was abducted by M_2 , but will not hold if an elopement occurred instead – and that is why the barred notation is used for this particular deleted clause. On the contrary, the fact $\text{current_place}(W, P_2)$, where P_2 is the home of M_2 , must necessarily be deleted by an effective execution of the operation, and so does not figure as barred.

The execution of plans is done through `assert` or `retract` commands on the facts to be, respectively, added or deleted. The plan's pre- and post-conditions are checked during the process, there being no effect in case of failure. A `log(L)` literal, initiated with $L=\text{start}$, is extended with each successful plan execution and can be usefully retrieved for a variety of purposes. On the basis of the log and of the initial state, which is saved when a session begins, it is possible to query about facts at any intermediate state. It is also possible to save and restore any previous state S (initial or intermediate), which enables simulation runs.

User interventions, necessary to achieve unplanned situations, are permitted in a limited scale through *directives* that can be either intermixed with the operations in a plan or called separately. Two of these are used in our example, one for changing *loves* facts, immune to the predefined operations, and the characters' beliefs, which may not correspond to actual facts.

To finish this partial review of the plan features, we remark that the planning algorithm $\text{plans}(G, P)$ is called in more than one way. More frequently G is given, as the goal, and P is a variable to which a generated plan will be assigned as output. However an inverse usage has been provided, wherein P is given and G is a variable; in this case, the algorithm will check whether P is valid and, if so, assign its net effects (a conjunction of F and $\text{not } F$ terms) to G .

4.2 The PlotBoard Tool

We shall briefly describe how **PlotBoard** works, after the controlling user, here called the Author, enters the `plot` command. The diagram of figure 3 will serve to guide the description.

The main option is to compose the plot from scratch, in a step-wise fashion. Ideally, the Author should leave a measure of autonomy to the characters (branching into the

² <http://www.swi-prolog.org/>

planner node of the diagram). At each step (cf. the `plan step` node), one subsequence of the plot will be generated. As if emerging from the mind of a character C , a short-term opportunistic goal G is instantiated by some goal-inference rule (C, S, G) , if the situation S of the rule currently holds. More than one rule may be simultaneously ready for activation, and the planning algorithm may find more than one alternative subsequence able to achieve the corresponding goals (whenever the planning algorithm backtracks), as indicates the self-loop around the `plan step` node. While a subsequence is presented, the Author is prompted to either issue an `ok` reply or call for an alternative, possibly after inspecting what effects it would have. An `ok` reply is followed by a return to the `planner` node.

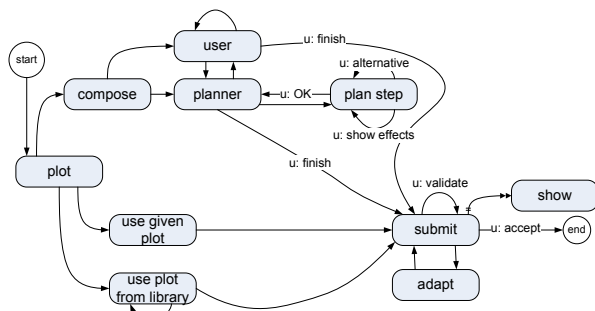


Figure 3: Flow of control of the **PlotBoard** prototype

The subsequence thus selected is then executed in a simulated mode, and the Author is asked whether the `plan step` iterations should continue, producing further subsequences to be appended to the plot so far obtained, or whether the composition process is finished for the time being (passing to the `submit` node), though still subject to possible adjustments.

If the Author is more inclined towards a closer arbitrary control than to the character autonomy policy described above, several options are available to determine the goals that the planning algorithm should try to achieve (cf. the the first 3 items of the menu for the `user` node below). Again, the self-loops around the `user` node represent the possibility of alternative plot subsequences being offered to the Author's choice. These options permit step-wise composition, which can be entirely commanded from the `user` node, but can also alternate with the activation of goal-inference rules, by intercalating transfers of control to the `planner` node.

An additional purpose of the `user` node is to prepare and support the composition process, by allowing to pose queries about the database state at each step, to change the operation level, and to issue directives to alter the characters' beliefs and the value of user-controlled properties.

- 1: goal
- 2: operation
- 3: list of operations
- 4: query
- 5: operation level
- 6: directive
- 7: planner
- 8: finish

Whatever composition policy is preferred – autonomous, arbitrary, or mixed – the finished plot is passed to the `submit` node. At this point, the Author can either accept the plot, which terminates the process, or can go through one or more rounds of adaptation, using the options offered at the `adapt` node below.

- 1: detail
- 2: summarize
- 3: change sequence
- 4: add operation
- 5: delete operation
- 6: replace operation
- 7: extend
- 8: queries or directives
- 9: insert motif
- 10: back to the submit options
- 11: stop

To help decide whether to accept the current plot or perform other adaptations, the `submit` menu allows to validate the plot (again through the planning algorithm). This may be in order if the Author directly introduces specific changes (items 4-6 of the `adapt` menu), noting that in all other forms of adaptation the planning algorithm intervenes to prevent integrity violations.

Another feature available at the `submit` node deserves attention, since what it produces, together with the menu-based dialogues, constitutes the intended output of **PlotBoard**. If selected, via the `show` option, it provides a visual display that can be repeated for the successive versions. For each operation in the current plot, the event it denotes appears as a rough drawing, side by side with a short template-driven natural language sentence.

We refer again to the diagram in figure 3, to consider two ways to obtain a plot without requiring step-wise composition from scratch. In both cases, a full plot is used to start with, and in both cases the process converges afterwards to the `submit` node.

Branching into `use given plot`, the Author can either enter the intended plot or retrieve a previously composed one. The planning algorithm is automatically called to inspect the plot, operation by operation, to check whether each of them can be applied in view of the pre- and post-conditions interplay. If an operation is found that can only be applied if a user-controlled property is tampered with, the possibility of changing the value of the property is indicated to the Author, who may or may not permit the execution of the necessary `vary` directive. If the Author denies permission, or if the offending property is not user-controlled, the plot is rejected.

In case the node `use plot from library` is chosen, the Library of Typical Plots (LTP) will be searched for items (S, G, P) , such that situation S currently holds, thereby propagating the instantiation of the parameter variables figuring in S to goal G and plot P . If more than one such item is found, the Author will have once more an opportunity to select the preferred P among the alternatives presented.

4.3 An example run

At the initial state, both Rama and Ravana are in their homes, respectively the royal palace and the city of Lanka, whereas Sita is alone in the forest. The two men love Sita, who only loves Rama. Starting to compose the plot, the Author invokes the planner in two stages, always selecting the detailed (level 2) alternatives. At this point the plot is, in natural language format:

Ravana rides from Lanka to forest. Ravana seizes Sita. Ravana carries Sita to Lanka. Rama rides from palace to Lanka. Rama defeats Ravana. Rama entertains Sita. Rama carries Sita to palace.

Wishing to try different versions, the Author looks at the `adapt` menu, shown in the previous section. The first change selected is the deletion of the two events that close the narrative. The next step is to issue directives to change

the emotional attachments and certain of the characters' beliefs: now Sita loves Ravana and Rama believes this fact. This justifies adding `entreat(Ravana, Sita)` as second event (after Ravana approaches the princess):

Ravana rides from Lanka to forest. Ravana entertains Sita. Ravana seizes Sita. Ravana carries Sita to Lanka. Rama rides from palace to Lanka. Rama defeats Ravana.

The plot now suggests the fake abduction pattern, wherein the villain seizes his pretended victim only to simulate a violent action. The Author wonders then if the same events could be arranged in some different sequence, and a dialogue ensues:

```
[f1:entreat(Ravana, Sita), f2:seize(Ravana, Sita)]
choose one of the fi tags: f1
[f1:seize(Ravana, Sita), f2:carry(Ravana, Sita, Lanka)]
choose one of the fi tags: f2
[f1:seize(Ravana, Sita), f2:ride(Rama, palace, Lanka)]
choose one of the fi tags: f1
Ravana rides from Lanka to forest. Ravana entertains Sita. Ravana carries Sita to Lanka. Ravana seizes Sita. Rama rides from palace to Lanka. Rama defeats Ravana.
```

This sounds as overt elopement after which the seducer restricts the woman's freedom. What can happen next?

Selecting the extend option of the adapt menu, the Author proposes: `current_place(Sita, palace)` as a goal, and the planner responds (figure 4) with: Rama captures Sita. Is this a satisfactory way to end the narrative? The Author selects option 8 and poses queries, to learn what the characters think and how they feel:

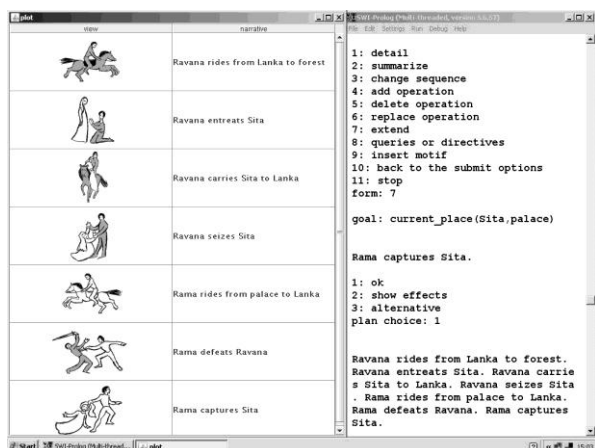


Figure 4: A PlotBoard screen.

```
query: beliefs
  Rama believes that Sita loves Ravana
  Ravana does not believe that Sita loves Rama
more queries?(yes/no): yes
query: emotional_condition
  Sita is bored. Rama is happy. Ravana is bored.
```

Sympathizing with the princess, the Author decides to revert the situation. Perhaps her love for the hero could revive (as happened with Helen of Troy), and the last event is replaced according to this expectation: `capture(Rama, Sita)` turns into `rescue(Rama, Sita)`. How does it look now? Back at the submit menu, the Author asks to visualize the scenes and accepts this result, a happy end for Sita as well as for the Author, who receives a grateful acknowledgement from the PlotBoard tool (Figure 5).

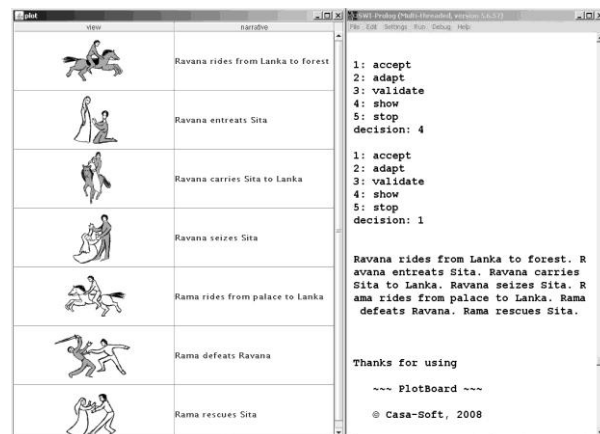


Figure 5: The accepted plot.

But much remains to be done. The *deus ex machina* directives should be replaced eventually by something internal to the narrative. Also, how to explain that Rama knew without being told that Sita had become Ravana's prisoner? To gather suggestions, to be possibly (re)used after due modifications appropriate to the genre, the Author might have inspected (figure 6) the applicable motifs, before issuing the final accept response, in which case the *life token*, the *love potion* (twice) and the *ordeal* motifs would be indicated at one or more positions in the plot wherein the respective motivating situation holds.

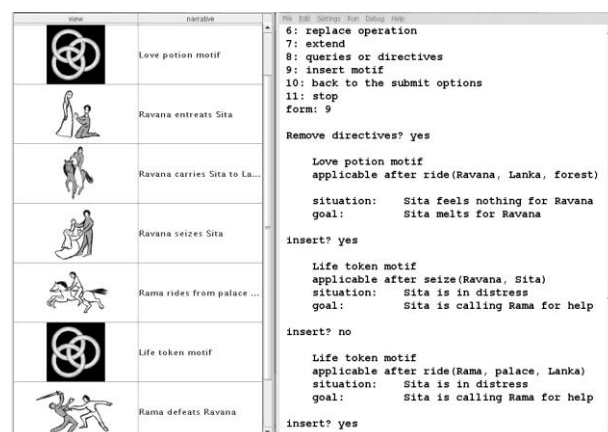


Figure 6: Insertion of motifs (partial view).

5. Concluding Remarks

Although the process of plot composition could surely be enriched far beyond what was presented here, the suggested fourfold approach seems to provide a sound initial basis. The conjecture that the interplay of the syntagmatic, paradigmatic, antithetic and meronymic relations already permits an ample coverage is reinforced by the connection between these relations and the four major tropes. Other concepts may be adduced to extend the model. If we see a disruption not as a discontinuity in one context, but as an attempt to put together two originally incompatible contexts, the notion of *blending* [Fauconnier & Turner; Casanova et al.] comes to mind, as the technique or art of conciliating the pending conflicts, often requiring a great deal of creativity.

The facilities associated with the four relations are adequate for other tasks, besides storyboarding, under suitable user interfaces. In interactive storytelling systems designed for entertainment, as well as in games, they might prove instrumental to support the production of

coherent stories with an ability to cause surprise. Further research might investigate ways to adjust the generation of alternatives to users' satisfaction models, so that there would be no longer a need to explicitly interfere to obtain varied and interesting outcomes.

Finally, let us recall that we have addressed the *fabula* level only, where one simply indicates *which* events should be included in the plots. A complex problem to be faced at the next level – the *story* level, where the concern is *how to tell* the events – is to find a plausible justification for the contextual disruptions introduced *ex machina* via user interaction. As said, such elaborations may be suggested by some fanciful motif annotated in the plot. Moreover a plurality of narrative objectives must be satisfied [Crawford; Turner; Montfort].

At the third and last level – the *text* level – the narrative is *represented* in some medium, not necessarily printed pages. Natural language text-generation from plots of log-registered business transactions is covered in [Furtado & Ciarlini 2000]. In the realm of literary genres, an ongoing project applies computer graphic animation to display narrative plots [Ciarlini et al.; Camanho et al.].

References

- AARNE, A. AND THOMPSON, S. 1987. *The Types of the Folktale*. Suomalainen Tiedeakatemia.
- ARISTOTLE. "Poetics". 2000. In *Classical Literary Criticism*. Penelope Murray et al. (trans.). Penguin.
- BAL., M. 2002. *Narratology*. U. of Toronto Press.
- BARROS, L. AND MUSSE, S. 2007. "Planning algorithms for interactive storytelling". In *ACM Computers in Entertainment*, ACM 5.1.
- BATINI, C., CERL, S. AND NAVATHE, S. 1992. Conceptual Design – an Entity-Relationship Approach. Benjamin Cummings.
- BÉROUL. 1970. *The Romance of Tristan*. A.S. Fedrick (trans.). Penguin.
- BREITMAN, K.K., BARBOSA, S.D.J., CASANOVA, M.A. AND FURTADO, A.L. 2007. "Using analogy to promote conceptual modeling reuse". In *Proc. of Workshop on Leveraging Applications of Formal Methods, Verification and Validation*.
- BREITMAN, K.K., CASANOVA, M.A. AND TRUSZKOWSKI, W. 2007. *Semantic Web*. Springer.
- BOOTH, W. 1974. *A rhetoric of Tropes*. U. of Chicago Press.
- BURKE, K. 1969. *A Grammar of Motives*. U. of California Press.
- CAMANHO, M., CIARLINI, A.E.M., FURTADO, A.L., POZZER, C.T. FEIJÓ, B. 2008. "Conciliating coherence and high responsiveness in interactive storytelling". In *Proc. 3rd. ACM International Conference on Digital Interactive Media in Entertainment and Arts (DIMEA 2008)*.
- CASANOVA, M.A., BARBOSA, S.D.J., BREITMAN, K.K. AND FURTADO, A.L. 2008. "Generalization and blending in the generation of entity-relationship schemas by analogy". In *Proc. of the Tenth International Conference on Enterprise Information Systems (ICEIS)*.
- CAVAZZA, M., CHARLES, F. AND MEAD, S. 2002. "Character-based interactive storytelling". *IEEE Intelligent Systems*, special issue on AI in Interactive Entertainment, 17(4).
- CHANDLER, D. 2007. *Semiotics: The Basics*. Routledge.
- CHRÉTIEN DE TROYES. 1983. *Le Chevalier de la Charrete*. M. Rocques (ed.). Honoré Champion.
- CIARLINI, A.E.M., POZZER, C.T., FURTADO, A.L. AND FEIJÓ, B., 2005. "A logic-based tool for interactive generation and dramatization of stories". In *Proc. of Advances in Computer Entertainment Technology*.
- COHEN, P.R. AND LEVESQUE, H.J., 1990. "Intention is Choice with Commitment". *Artificial Intelligence* 42.
- COSTIKYAN, G. 2002. "I have no words and I must design: Toward a Critical Vocabulary for Games". In *Proc. of Computer Games and Digital Cultures*.
- CRAWFORD, C., 1984. *The Art of Computer Game Design*. Osborne-McGraw-Hill.
- CULLER, J. 1983. *On Deconstruction: Theory and Criticism after Structuralism*. Cornell U. Press.
- FAUCONNIER, G. AND TURNER, M. 2002. *The Way We Think*. Basic Books.
- FIKES, R.E. AND NILSSON, N.J. 1971. "STRIPS: A new approach to the application of theorem proving to problem solving". *Artificial Intelligence* 2.
- FURTADO, A.L., CASANOVA, M.A., BARBOSA, S.D.J. AND BREITMAN, K.K. 2008. "Analysis and Reuse of Plots using Similarity and Analogy". In *Proc. of 27th International Conference on Conceptual Modeling (ER)*.
- FURTADO, A.L. AND CIARLINI, A.E.M. 2000. "Generating Narratives from Plots using Schema Information". In *Proc. of 5th International Conference on Applications of Natural Language to Information Systems*.
- FURTADO, A.L. AND CIARLINI, A.E.M. 2001. "Constructing Libraries of Typical Plans". In *Proc. of 13th International Conference on Advanced Information Systems Engineering*.
- FURTADO, A.L. AND VELOSO, P.A.S. 1996. "Folklore and Myth in The Knight of the Cart". In *Arthuriana*, vol 6, 2.
- GRASBON, D. AND BRAUN, N. "A morphological approach to interactive storytelling". 2001. In *Proc. CAST01, Living in Mixed Realities*. Special issue of *Netzspannung.org/ journal*, the Magazine for Media Production and Inter-media Research.
- HEARST, P.C. AND MOSCOW, A. 1988. *Patty Hearst: her own Story*. Avon.
- HOMER. *The Iliad*. 1950. E.V. Rieu (trans.). Penguin.
- KAUTZ, H. A. 1991. "A Formal Theory of Plan Recognition and its Implementation". In *Reasoning about Plans*. J. F. Allen et al. (eds.). Morgan-Kaufmann.
- LAKOFF, G. AND JOHNSON, M. 1980. *Metaphors We Live By*. U. of Chicago Press.
- LLOYD, W. 1987. *Foundations of Logic Programming*. Springer.
- MARCHELLO-NIZIA, C. (org.). 1995. *Tristan et Yseut*. Gallimard.
- MATEAS, M., AND STERN, A. 2000. "Towards integrating plot and character for interactive drama". In *Socially Intelligent Agents: the Human in the Loop*, AAAI Fall Symposium.
- MCGARRY, M. (ed.). 1979. "The Story of Deirdre". In *Great Folk Tales of Ireland*. Frederick Muller.
- MONTFORT, N. 2006. "Natural Language Generation and Narrative Variation in Interactive Fiction". In *AAAI Workshop on Computational Aesthetics*.
- ORTONY, A. (ed.). 1996. *Metaphor and Thought*. Cambridge: Cambridge U. Press.
- OVID, 1986. *Heroides and Amores*. G. Showerman (trans.). Harvard U. Press.
- PEARCE, C. (2002) "Emergent authorship: the next interactive revolution". In *Computers & Graphics* 26.
- PROPP, V. 1968. *Morphology of the Folktale*. S. Laurence (trans.). U. of Texas Press.
- RAO, A.S. AND GEORGEFF, M.P. 1991. "Modeling rational agents within a BDI-architecture". In *Proc. of Int'l Conf. on Principles of Knowledge Representation and Reasoning*.
- RIEDL, M. and YOUNG, R. M. 2004. "An intent-driven planner for multi-agent story generation". In *Proc. of 3rd Int'l. Conf. on Autonomous Agents and Multi Agent Systems*.
- SAUSSURE, F. 2006. *Cours de Linguistique Générale*. C. Bally, A. and A. Riedlinger (eds.). Payot.
- TITUS LIVIUS. 1919. *History of Rome*, vol I. B. O. Fster (trans.). Harvard U. Press. Book I, Chapter I: 13.
- TRUONG, K. N. HAYES, G. R. AND ABOWD, G. D. 2006. "Storyboarding: An Empirical Determination of Best Practices and Effective Guidelines". In *Proc. of 6th conference on Designing Interactive systems*.
- TURNER, S.R. 1992. *Minstrel: A computer model of creativity and storytelling*. T. R. UCLA-AI-92-04, Computer Science Dept.
- VALMIKI. 1999. *Le Ramayana*. Philippe Benoit et al. (trans.). Gallimard.
- VIRGIL. 1994. *Eclogues, Georgics, Aeneid*. H.R. Fairclough (trans.). Harvard U. Press.
- WALLIS, J. 2008. *Making Games that Make Stories*. Electronic book review [www.electronicbookreview.com].
- WARDRIE-FRUIIN, N., HARRIGAN, P. (eds.). 2004. *First Person: New Media as Story, Performance, and Game*. The MIT Press.
- WARREN, D.H.D. 1974. *WARPLAN: a System for Generating Plans*. Edinburgh: University of Edinburgh, Department of Computational Logic, memo 76.
- WHITE, H. 1973. *Metahistory: The Historical Imagination in Nineteenth-Century Europe*. The Johns Hopkins U. Press.