Creating Dynamic Story Plots with Continual Multiagent Planning

Michael Brenner
Albert-Ludwigs-Universität
Freiburg, Germany
brenner@informatik.uni-freiburg.de

Abstract

An AI system that is to create a story (autonomously or in interaction with human users) requires capabilities from many subfields of AI in order to create characters that themselves appear to act intelligently and believably in a coherent story world. Specifically, the system must be able to reason about the physical actions and verbal interactions of the characters as well as their perceptions of the world. Furthermore it must make the characters act believably–i.e. in a goal-directed yet emotionally plausible fashion. Finally, it must cope with (and embrace!) the dynamics of a multiagent environment where beliefs, sentiments, and goals may change during the course of a story and where plans are thwarted, adapted and dropped all the time. In this paper, we describe a representational and algorithmic framework for modelling such dynamic story worlds, Continual Multiagent Planning. It combines continual planning (i.e. an integrated approach to planning and execution) with a rich description language for modelling epistemic and affective states, desires and intentions, sensing and communication. Analysing story examples generated by our implemented system we show the benefits of such an integrated approach for dynamic plot generation.

Introduction

To tell a story is a challenging task that involves many (if not most) aspects of human intelligence. If the storyteller is an AI system it must effectively simulate a coherent story world and control a number of virtual characters in a manner that seems believable to humans, much as in the Turing Test. Thus, creating believable stories, whether interactively or not, can be considered an "AI-complete" task and requires methods from many subfields of AI, e. g., planning, virtual agents and multiagent systems, reasoning about beliefs and intentions, affective computing, and dialogue systems.

Among these methodologies, planning has probably received the most attention in story generation (Meehan 1977; Lebowitz 1985; Riedl and Young 2004; Si, Marsella, and Riedl 2008). Its relevance results from the structural similarity between plans and plots, both of which describe temporal and causal relations between events. Indeed, temporal-causal coherence, as modeled by the semantics of classical STRIPS-like planning formalisms, can be considered a necessary condition for stories (at least non-postmodern ones). Yet, Planning research follows a different research agenda than Narrative Intelligence and therefore has developed representations and algorithms that are only of limited use to plot creation. As a result, while plans are used in storytelling systems, many interesting aspects of narrative, e. g., motivation and emotion, must be handled outside the planner. While this is fine in itself, it prevents long-time plotting, usually done by a planner, from taking these aspects into account, e. g. say, planning for motivation changes based on emotional reactions to events. Therefore, in this work, we try to integrate ideas from many of the AI fields mentioned above directly into the planning formalism and algorithm to make it directly suitable for narrative generation.

The paper is structured as follows. We first review relevant work in fields outside narrative research. Then we describe a planning formalism that integrates many of these relevant aspects. We then describe a planning algorithm using this representation, Continual Multiagent Planning, and briefly present our implementation. Analysing a story generated by our program we discuss the benefits of our approach. We conclude with a discussion of its relation to previous work in narrative generation and its possible future uses and extensions.

Related Work I

Our work integrates ideas from several subfields of AI, in particular classical and distributed planning, multiagent systems, knowledge representation and reasoning (mainly about perceptions, beliefs, and emotions) and dialogue systems. Due to space limits, we can only discuss few prototypical inspirations here. At the end of the paper, we will relate our approach to previous work in storytelling research.

Most stories feature several characters and can thus be regarded as *multiagent* environments. To model the beliefs of different agents (and their reasoning about each other) we will integrate multiagent *epistemic* modalities into the planning representation (Fagin et al. 1995). Additionally, similarly to *BDI* models of multiagent cooperation, we will explicitly model the desires and intentions of different agents (Grosz and Kraus 1996). In order to describe how characters gather new information, we will need to model communication and perception as well. Here, we are inspired by approaches to *collaborative dialogue* (Lochbaum 1998) and planning with *sensing* (Petrick and Bacchus 2002).

Algorithmically, our work is based on the intuition that the dynamics of plots are hard to describe naturally with a single plan. Often, plots depend on plans failing or being thwarted, then being dropped or adapted, and finally succeed or fail (where which is which often lies in the eye of the beholder), i.e. as a series of planning and execution cycles performed by the characters. Therefore, what we develop is a *Distributed Continual Planning* (DCP) method (DesJardins et al. 1999; Brenner and Nebel 2009).

Modelling Story Worlds in a Multiagent Planning Formalism

Story worlds usually are multiagent environments. To describe plausible behaviour in such worlds, we need to reason about their dynamics. Thus, we must be able to represent not only the physical actions that agents can perform, but also their perceptual and communicative capabilities as well as their (mutual) beliefs and (joint) goals. To do this in a domain-independent fashion, we use a formal descripition language, the Multiagent Planning Language MAPL (Brenner and Nebel 2009). In this section, we present MAPL informally and discuss the extensions made for this paper.

MAPL is a multiagent variant of PDDL (Planning Domain Definition Language), the de facto standard language for classical planning. Instead of propositions, MAPL uses multi-valued state variables (MVSVs). For example, a state variable *color(ball)* would have exactly one of its possible *domain* values *red, yellow,* or *blue*, as compared to the three semantically unrelated propositions *(color ball red), (color ball yellow), (color ball blue)*. MVSVs have successfully been used in classical planning in recent years,but they also provide distinct benefits when representing story worlds: a) Incomplete knowledge can be expressed by adding an *unknown* value to the domain of a MVSV, b) this idea can be extended to model beliefs and mutual beliefs among characters (Fagin et al. 1995) c) knowledge-seeking actions can be modelled as supplying a yet unknown MVSV value. Thus, sensing and communicative acts are modelled like *wh-questions* (what colour? where? etc.), d) due to the mutual exclusivity of MVSV values, they are well suited for Initial State Revision (Riedl and Young 2006).

In addition to the usual preconditions and effects, MAPL actions have a **controlling agent** who executes the action. MAPL assumes that the controlling agents are fully autonomous when executing actions, i. e. there is no external synchronization or scheduling component. Consequently, an action will only be executed if, in addition to its preconditions being satisfied, the controlling agent *knows* that they hold. Implicitly, all MAPL actions are extended with such **knowledge preconditions**. Similarly, implicit **commitment preconditions** describe, intuitively, that if action a controlled by agent A is included in agent B's plan, this can only be done if A has agreed to perform a.

MAPL models three different ways to affect the beliefs of agents: sensing, copresence, and communication. **Sensor models** describe under which conditions the current value of a MVSV can be perceived. **Copresence models** are multiagent sensor models that induce *mutual belief* about the perceived state variable. Informally, agents are copresent when they are in a common situation where they can not only perceive the same things but also each other. **Communicative acts** in MAPL include declarative statements, questions, requests, and acknowledgments. While declarative statements change the belief state of another agent similarly to sensory actions, the other types of communicative acts affect aspects of the agent that are typically considered static in AI Planning, namely the goals of agents.

MAPL **goals** are first-order formulae, like in PDDL. For storytelling we mostly use them in a specific *conditional* form: By introducting a new MAPL keyword, **"currently"**, we can refer to the current state of the world and the agents' beliefs about it in such a conditional goal formula. MAPL also has **temporary subgoals** (TSGs), which must be satisfied at some point in the plan, but may be violated in the final state. TSGs are represented as explicit symbols in MAPL and thus can be reasoned about by a planner. In particular, they can be active or inactive. This is also true for conditional goals, whose antecedent (condition) may hold in the current state or not. Both kinds of *goal activation* mimic how commitment turns desires into intentions in BDI models of human practical reasoning (Bratman, Israel, and Pollack 1988; Cohen and Levesque 1990). Throughout, the paper we will often refer to activated goals as intentions. **Assertions** are *counterfactual* statements, e. g., "If I knew where to find a sword, I could slay the dragon", that the continual planner may use as temporary subgoals in order to gather missing information necessary to achieving its main goal. Assertions enable the agent to postpone planning for subproblems until it has gained more knowledge, i. e. by partially executing a plan and then switching back to more detailed planning. Thus, assertions encourage proactive goal-driven information gathering (Brenner and Nebel 2009), which for plot generation often seems to be a desirable character trait.

MAPL plans are partially ordered, using different kinds of *causal links*. This is advantageous for plot generation because plans provide explanations for the behaviour of the characters. In contrast to other plan-based approaches we will not use plans directly to represent the whole plot. Since during a continual planning episode usually multiple plans are being generated, executed, and revised, we consider as the plot the execution history of the episode, annotated with relevant (possibly false) beliefs and goals. This **plot graph** comprises a totally ordered "fabula", i. e. the sequence of events that occur. Yet, it also uses explanations from plans and plan monitoring to relate actions to each other by various types of causal links. Such causally annotated histories can be naturally regarded as plots in the sense of E. M. Forster (Forster 1927) and provide ample information for discourse generation, i. e. the presentation of the plot.

Continual Multiagent Planning

How can we generate MAPL plot graphs? The method presented in this section, Continual Multiagent Planning (CMP), is a distributed algorithm, i. e. it describes planning *by* multiple agents, who all have different motivations and beliefs about the world. Being fully distributed, it can be applied to large interactive environments, e. g., multiplayer online role-playing games. However, it also models planning *for* multiple agents, since even the individual agents' plans may involve several agents if that is necessary to achieve her goals. For storytelling, this means that CMP allows for both character-centric and author-centric plot generation.

The CMP algorithm is shown in algorithm 1 (its subprocedures will only be presented informally). CMP extends Continual Collaborative Planning (CCP), an approach developed in the context of situated human-robot interaction (Brenner and Nebel 2009). Like in CCP, CMP agents deliberately switch between planning, (partial) plan execution, monitoring, plan adaptation and communication. However, CMP agents are not required to be benevolent and always willing to adopt any TSGs proposed by other agents – luckily, since this would prevent conflict, intrigue and drama in the generated plots.

Algorithm 1 CMP AGENT(S, G)

```
P = ∅
Received no message:
  if S satisfies G do
    return "goal reached"
  else
    P = MONITORINGANDREPLANNING(S, G, P)
  if P = ∅ then
    return "cannot achieve goal G"
  else
    e = EXECUTENEXTACTION(S, P)
    (S, P) = STATEESTIMATION(S, e)
Received (tell-val vx) from agent a:
  add v ≐ x to S
Received request(sg) from agent a:
  if cooperative(self, a) ∉ S then
    send "will not adopt request sg" to a
  P = MONITORINGANDREPLANNING(S, G ∪ sg, ∅)
  if P = ∅ then
    send "cannot execute request sg" to a
  else
    add sg to G as temporary subgoal
    send "accept request sg" to a
```

When used for centralised planning (i. e. one planner controls all agents) or when no communication is taking place, a CMP agent alternates between (re-)planning and execution. Subprocedure MONITORINGANDREPLANNING first determines whether a new planning phase should be triggered, either because the agent has sensed an external event that has invalidated its previous plan, or because her goals themselves have changed, or because of an *assertion* that was previously used to advance planning despite missing information and whose detailed planning is now triggered because additional knowledge has become available (Brenner and Nebel 2009). If, for any of the above reasons, planning is triggered the agent replans for those parts of its plan that are no longer valid. The details of the actual (re)planning are irrelevant for the purpose of this paper (any state-of-the-art PDDL planner may be adapted for the purpose); it results in an asynchronous MAPL plan that specifies actions for (possibly) several agents and the causal and temporal relation between them necessary for achieving the planning agent's goal. If the plan involves other agents than the planning agent or those characters she can directly control, the new plan must ensure that they are commited to the (sub)goals their actions contribute to. In the original CCP the new plan would have included maximally one *negotiate_plan(a)* action for each agent a appearing in the plan, since all agents were supposed to be cooperative and their exact role in the plan could freely be discussed in the negotation phase. This is different in CMP, where the planning agent must consider different options for making a commit to a subgoal. This can either be done by negotiation as before, if a is known to be cooperative, or by some form of persuasion, i. e. indirect activation of a conditional goal of a. For example, a hunter may present a bear with a honey comb to raise its appetite and make it walk into a trap. Indirect activation may also be recursive, e. g., when a bank robber r threatens a bank clerk c, thereby making *cooperative(c,r)* true and thus make c open for "requests" in the next step.

As soon as a CMP agent has found (or repaired) a valid plan it enters the execution phase (function EXECUTENEXTACTION). First, an action e on the first level of the plan, i. e. one whose preconditions are satisfied in the current state, is chosen non-deterministically. If the action is executable by the CMP agent himself (this includes communicative actions), it is executed. If not, the planning agent tries to determine whether the action was executed by its controlling agent, i. e. it actively observes changes in the environment relevant for its plans. In both cases, the CMP agent will try to update its knowledge about the world state based on the expected effects and the actual perceptions made (function STATEESTIMATION).

Implementation In our view plots do not only consist of plans, but also of their execution, and the resulting re-evaluation of beliefs, goals, and plans by all character agents. Such an approach can best be implemented in a simulation environment. This is most obvious in *interactive* narrative, where some characters are not controlled by the system, but by human users. Yet simulation is also a convenient way to compute the complex results of several characters acting simultaneously in a common environment, observing and influencing each other constantly, even if controlled by a single "author". Therefore we have implemented MAPSIM, a software environment that automatically generates multiagent simulations from MAPL domains. In other words, MAPSIM interprets the MAPL domain both as the planning domain for each CMP character, but also as an executable model of the environment, so that it can determine the results of the execution of the characters' actions.

Note that while for generating the story analysed in the following section, we invoked MAPSIM non-interactively to emphasise the autonomy of the approach, MAPSIM can also be used interactively. Human users may "play" the role of any character and send MAPL commands to the simulation directly.

Figure 1: A story in the *Quests* domain non-interactively created by MAPSIM.

> 1 This is a story about Smaug, King Arthur and Prince Valiant.
>
> 2 King Arthur was in the castle. 3 The treasure was in the cave. 4 King Arthur rode to the cave. 5 King Arthur saw that Smaug was in the cave.
>
> 6 King Arthur rode to the castle. 7 King Arthur saw that Prince Valiant was in the castle. 8 'Please bring me the treasure, Prince Valiant,' King Arthur said. 9 'As you wish, King Arthur,' Prince Valiant replied. 10 'Where is the treasure, King Arthur?' Prince Valiant asked. 11 'The treasure is in the cave, Prince Valiant,' King Arthur said. 12 'Thank you,' Prince Valiant said.
>
> 13 Prince Valiant rode to the cave. 14 Prince Valiant saw that Smaug was in the cave. 15 Smaug tried to kill Prince Valiant - but failed! 16 Prince Valiant saw that Smaug was not dead. 17 Prince Valiant killed Smaug.
>
> 18 Prince Valiant took the treasure. 19 Prince Valiant rode to the castle. 20 Prince Valiant gave King Arthur the treasure. 21 'Thank you for bringing me the treasure, Prince Valiant,' said King Arthur.
>
> 22 King Arthur and Prince Valiant lived happily ever after. Smaug did not.

Analysis of a Worked Example

In order to show the benefits of our integrated approach, we will now analyse a story generated by MAPSIM. It is reproduced in figure 1. During the creation of the story a total of 20 different plans were created by the three characters and the simulation itself. On a 1.6 GHz Intel Pentium and 1GB RAM the *total* planning time was less than 500ms.

As input, MAPSIM was given a formal MAPL domain description and individual goals and beliefs for each of the three characters as well as the true initial world state. The resulting plot graph is (for reasons of space) only shown partly in figure 2. It roughly corresponds to lines 9–15 of figure 1 and gives an impression of the MAPL representation of the plot. The first and last line of figure 1 have no direct correspondence in the plot graph, but are directly produced by MAPSIM: In line 1 the characters are introduced, whereas in line 22 MAPSIM reports on which of the agents have achieved their goal and which have not.[1]

Multimodal interaction Note first that characters' behaviour as generated by CMP seamlessly interleaves physical action, sensing, and communication, e. g. in lines 6–8. Due to the explicit representation of epistemic states and information-gathering actions, the characters will plan which gaps in their knowledge they need to fill in order to further detail their plans. This may result in active observation (as in line 5, where Arthur checks whether the cave is empty) or in information-seeking *subdialogues* (as in lines 10–12).

Plan dynamics When Arthur arrives at the cave, he observes that the dragon, Smaug, is there. Arthur knows that he cannot take the treasure while the dragon is present. Thus, CMP detects, in its monitoring phase, that Arthur's plan has become invalid. Arthur generates a new plan, this time a multiagent plan in which Valiant is supposed to help him get the treasure. Switching to the new plan, Arthur leaves the cave and returns to the castle. We claim that it would be quite difficult to describe the plot so far with a single plan, let alone generate it with a single planner run. Continual planning, on the other hand, seems like the natural way to model how a character reacts to the obstacles she encounters.

A form of *proactive* continual planning is exemplified in lines 8-13. Prince Valiant initially does not know the location of the treasure. Thus he could normally not find a plan to get it and therefore would have to decline Arthur's request. However, the planning domain contains a counterfactual *assertion* stating, informally: "If I knew where the treasure was, I could make a plan to bring it somewhere else". Using this assertion, Valiant is able to deliberately *postpone* part of the planning process and first engage in the short subdialogue of lines 10–12 in order to gather the missing information (Brenner and Nebel 2009). The semantics of assertions is such that, when the missing information becomes available, a new planning phase is triggered. It provides Valiant with a more detailed plan – which he executes until he also encounters Smaug and must again extend the plan to include fighting the dragon. In a different setting ("If I knew where the grail was...") satisfying the replanning condition of the assertion, i. e. closing the knowledge gap, may be the more complex part and constitute most of the resulting plot.

Goal dynamics The standard use of continual planning is to adapt plans to changing conditions in the outside world or an agent's belief about it. However, in real life (and thus in stories) *motivations* change, too. CMP agents can adopt temporary subgoals, e. g., when accepting a request by another agent, as in lines 9 and 12 of figure 1. Such changing goals usually lead to more substantial changes in the plot than mere plan adaptation for the same goal. Only after Arthur's request (line 8), Valiant gets involved in the story at all. In particular, this prompts a nice example of *mixed-initiative* behaviour (line 10), where Valiant immediately asks back to get more information necessary to achieve his new goal.

Characterisation by affective goal activation As noted above, changes in an agent's goals may lead to substantial changes in her behaviour. Indeed, it can be argued that a character is better characterised by her motivations than her (fairly volatile) current state of knowledge. However, a complex character has many motivations. Depending on her internal (or some external) context, motivations may be active (i. e. they drive her current behaviour) or inactive. Such context-dependent *goal activation* allows for a more fine-

[1] Obviously the textual output could be vastly improved, e. g., by proper generation of referring expressions. This output was generated using action-specific English templates that the MAPL domain can be annotated with. This way, plot graphs from arbitrary domains can quickly be rendered into a readable natural-language output.

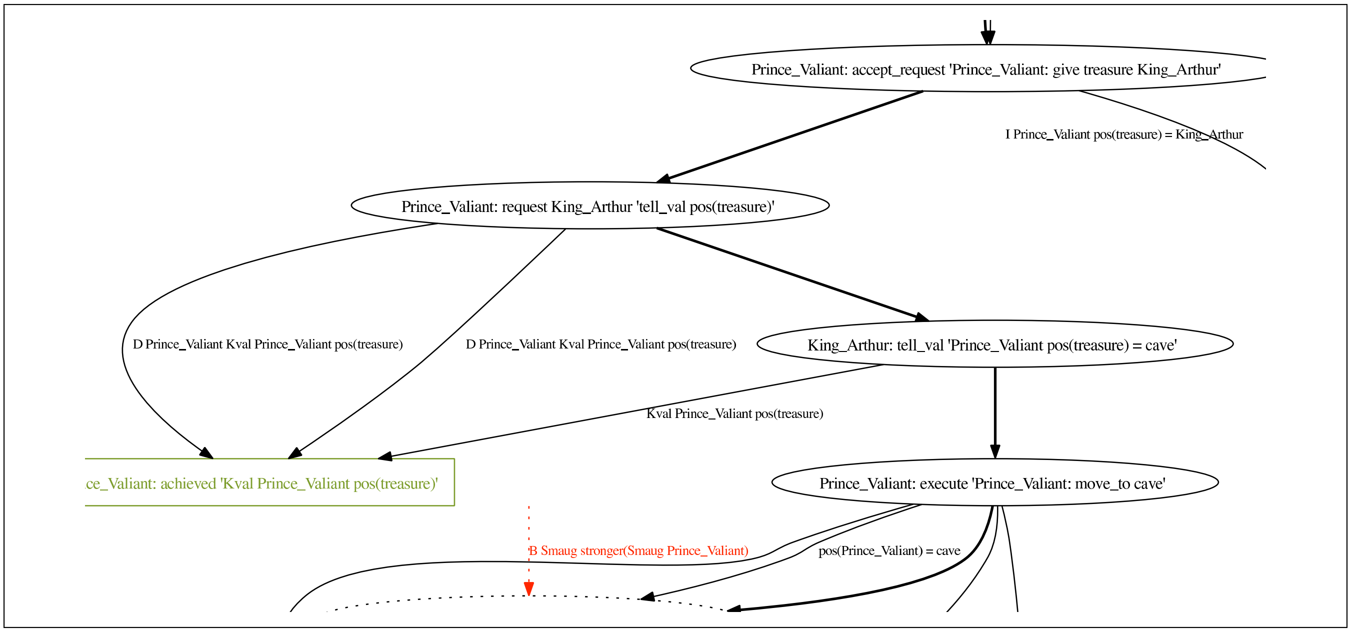

Figure 2: Plot graph (excerpt) for the *Quest* story generated by MAPSIM. Legend: temporal links (bold black), causal links (black), threat prevention links (blue), false belief links (red), TSG achievement (olive box). For clarity, causal links with facts that are true at the beginning have been omitted.

grained characterisation. e. g., in our story, the dragon will only want to kill humans when they have entered its lair.

Using MAPL's *currently* keyword, we can refer to the current state in a goal formula and describe the conditions for goal activation and deactivation.Several such conditional goals can be defined for a character. Together they can define a multi-faceted personality whose concrete intentions may change depending on the current situation, but who will show consistent behaviour in similar situations.

It is important for storytelling that the conditional goals characterising an agent can refer to *emotions* or mental attitudes directed towards other agents and objects, e. g., `angry(a)`, `loves(a,b)`, etc. For example, if the dragon only attacked intruders when angry, but was willing to share the treasure when happy, another story might tell how Prince Valiant charmed the dragon and thus could acquire the treasure without violence. This also opens CMP for use in affective storytelling (Pizzi and Cavazza 2007).

Beliefs, desires, intentions Through MAPL's commitment preconditions CMP enforces characters to commit to a goal/desire before they can actively pursue it, i.e. make it an intention first (Bratman, Israel, and Pollack 1988; Cohen and Levesque 1990). In the multiagent case this means that if character A is not directly controllable by another agent B, B must first somehow persuade A to commit to a new goal before B can assume A's working towards it. In our story, Arthur knows that he cannot "use" Valiant in his plan to get the treasure unless Valiant commits to that goal himself, i. e. makes it an intention of his own. Here, CMP finds a plan for Arthur to achieve this using a simple request (lines 8–9), since in the MAPL description Valiant has been modelled as being cooperative towards Arthur. On the other hand, before CMP could include actions of the dragon into Arthur's plans, it would first have to indirectly activate one of the dragon's own desires.

False beliefs are important for plots, as they result in misunderstandings, in misguided or unsuccessful behaviour. Again, continual planning can be used to reason about the consequences of believing something wrongly. To show this, the example domain is set up such that the "stronger" character always wins a fight. Here, Smaug falsely believes to be stronger than Prince Valiant and attacks him (line 15), which eventually leads to his own death.

Chekhov's gun The plot graph describes which facts and objects are used in the plot. Those facts should be mentioned so that the reader/player can follow the reasoning of the characters. Crucially, the plot graph does not only point to preconditions of actions that characters actually perform, but also to those beliefs never used in an executed plan (because of the plan or the belief becoming obsolete first), but that are necessary to *explain* the changing motivations and plans of the characters.

Related Work II

Having presented our approach to planning for storytelling, we can finally relate it to existing storytelling systems (again, only few representative ones). It should be kept in mind, though, that we do not claim this to be a full storytelling framework, but only to provide planning representations and algorithms appropriate for being used *inside* such a system. MAPSIM is only a research prototype and a domain-independent testbed to evaluate variants of CMP.

CMP, when used by individual agents in a multiagent simulation like MAPSIM, works as a character-centric approach to storytelling (in contrast to author-centric and story-centric approaches (Mateas and Sengers 1999)) in the tradition of Meehan's Tale-Spin (Meehan 1977). However, since the intentions of several characters are reasoned about explicitly and individually in each plan, it also integrates aspects of author-centric approaches. In this respect, our approach seems closest to Fabulist (Riedl and Young 2004). When CMP is used in the context of a simulation its capability to deliberately devise plans that may fail and to reason about dynamic goals of characters makes it quite suitable to be used for dynamic, interactive storytelling like Emergent Narrative (EM) (Aylett 1999).

Thus, MAPL and CMP integrate features of both author-centric and character-centric approaches. It would be of great interest to evaluate their use in Interactive Storytelling frameworks that strive for a similar integration, e. g., (Si, Marsella, and Riedl 2008; Porteous and Cavazza 2009).

Discussion

The main contribution of this article is an integration of models and methods from a number of AI subfields into a representational (MAPL) and algorithmic (CMP) framework that is well-suited for an "AI-complete" task like storytelling. Providing both a rich representation language for reasoning *about* multi-character environments and a distributed algorithm for planning *by* multiple agents, it combines aspects of both character-centric and author-centric approaches to storytelling. A specific emphasis has been put on enabling the generation of *dynamic* plots, in which beliefs, motivations and character traits may change. We believe that, due to a decade-long focus on planning as an isolated one-shot problem, these dynamics have been insufficiently studied in both planning and storytelling research—despite the fact that plot twists and character development are often said to be what makes a story interesting.

Interestingness or, more technically, *plot quality* is not an explicit concept anywhere in our approach. Thus it is not surprising that we cannot reliably claim that our approach will generate interesting stories. To this end, we will have to extend the approach by an author or, even better, a reader model. In future work, we will investigate how a CMP *author agent* can try to achieve *plot goals* by means of initial state revision (Riedl and Young 2006) and late commitment (Swartjes and Vromen 2007), concepts inspired by game-mastering in pen-and-paper roleplaying games. CMP supports these ideas almost directly, because it can reason about possible MVSV values that have not yet been sensed by any of the characters. In further work, we will consider *reader agents* who follow the unfolding of the plot and assess it using subjective, externally defined quality metrics. Given such a metric, we can iteratively use the author agent to generate series of slightly modified stories, i. e. we can mimic the process of *story revision* in a form of *local search* in the space of plot graphs.

At first glance, creating believable interactions between characters in a story world may seem remote from more "serious" AI topics like robotics. However, this work was inspired by our research on human-robot collaboration and will, in turn, most certainly find itself being integrated into robots again now.

Acknowledgments

This research was supported by the EU as part of the Integrated Project CogX (FP7-ICT-2xo15181-CogX).

References

Aylett, R. 1999. Narrative in virtual environments - towards emergent narrative. In *Proc. AAAI Narrative Intelligence Symposium*.

Bratman, M. E.; Israel, D. J.; and Pollack, M. E. 1988. Plans and resource-bounded practical reasoning. *Computational Intelligence* 4:349–355.

Brenner, M., and Nebel, B. 2009. Continual planning and acting in dynamic multiagent environments. *Journal of Autonomous Agents and Multiagent Systems* 19(3).

Cohen, P. R., and Levesque, H. J. 1990. Intention is choice with commitment. *Artificial Intelligence* 42(213–261).

DesJardins, M.; Durfee, E.; C. Ortiz, J.; and Wolverton, M. 1999. A survey of research in distributed, continual planning. *The AI Magazine*.

Fagin, R.; Halpern, J. Y.; Moses, Y.; and Vardi, M. Y. 1995. *Reasoning About Knowledge*. MIT Press.

Forster, E. M. 1927. *Aspects of the Novel*. San Diego: Harcourt.

Grosz, B. J., and Kraus, S. 1996. Collaborative plans for complex group action. *Artificial Intelligence* 86.

Lebowitz, M. 1985. Story-telling as planning and learning. *Poetics* 14:483–502.

Lochbaum, K. E. 1998. A collaborative planning model of intentional structure. *Computational Linguistics*.

Mateas, M., and Sengers, P. 1999. Narrative intelligence. In *Proc. Narrative Intelligence Symposium*.

Meehan, J. R. 1977. Tale-spin, an interactive program that writes stories. In *Proc. IJCAI*.

Petrick, R., and Bacchus, F. 2002. A knowledge-based approach to planning with incomplete information and sensing. In *Proc. AIPS-02*.

Pizzi, D., and Cavazza, M. 2007. Affective storytelling based on character's feeling. In *Proc. AAAI Symposium on Intelligent Narrative Technologies*.

Porteous, J., and Cavazza, M. 2009. Controlling narrative generation with planning trajectories: the role of constraints. In *Proc. ICIDS*.

Riedl, M. O., and Young, R. M. 2004. An intent-driven planner for multi-agent story generation. In *Proc. AAMAS*.

Riedl, M. O., and Young, R. M. 2006. Story planning as exploratory creativity: Techniques for expanding the narrative search space. *New Generation Computing* 24(3).

Si, M.; Marsella, S. C.; and Riedl, M. O. 2008. Integrating story-centric and character-centric processes for authoring interactive drama. In *Proc. AIIDE*.

Swartjes, I., and Vromen, J. 2007. Emergent story generation: Lessons from improvisational theater. In *Proc. AAAI Fall Symposium on Intelligent Narrative Technologies*.