

Towards an Author Intention Based Computational Model of Story Generation

Feng Zhu^{1,2}, Cungen Cao¹

¹Key Laboratory of Intelligent Information Processing, Institute of Computer Technology,
Chinese Academy of Science, Beijing, China

freecafeabc@gmail.com, cgcao@ict.ac.cn

²Graduate University of Chinese Academy of Science, Beijing, China

Abstract. This paper addresses the problem of plot controllable story generation model. Since most of the previous works focus on logically flawless plot, the main challenge is the need for more generic method to generate dramatic and interesting plot. Motivated by this background, this paper proposes a computational method for plot controllable story generation. Firstly, we use planning as a model of plot generation. Then we utilize author intentions as plot constraints to force the planner to consider substantially more complex plans. Finally we integrate author intentions into planning and develop a plot controllable Graphplan algorithm. Experimental results demonstrate the effectiveness of our approach.

Keywords: story generation, narrative generation, author intention, graph plan

1 Introduction

Story as entertainment, in the form of oral, written, or visual storytelling, plays a central role in many forms of entertainment media, including novels, movies, television, and theatre. Since the emergence of the first computational story generation system, TALE-SPIN, developed by Meehan in 1976, automated story generation has been extensively studied, with applications ranging from computer games to education and training [1-3]. While a majority of these studies are on automatic generation of logically flawless plot, the artistic properties of story, such as plot controllability, which is an essential story element for the reader's enjoyment, has received less attention. This paper addresses one of the central problems in the automatic creation of plot controllable story, which keeps the reader engaged in the interesting plot, giving them high entertainment value.

Since story plot can be naturally modeled as a sequence of actions, AI planning has emerged as the technology of choice in the field and a range of planning approaches have been successfully applied to the task of story generation. However the characteristics of a "good" plan, such as optimality, aren't necessarily the same as those of a "good" story, where convoluted or redundant plot sequences may offer more reader interest. For example, if a planner were given an initial state in which a character was rich and an outcome state in which the character was rich, the planner would simply indicate that there was no problem to solve. While a character that begins rich, then

becomes poor, and finally regains the state of being rich may be an interesting plot. We argue that this limits the applicability of off-the-shelf planners as techniques for creating stories.

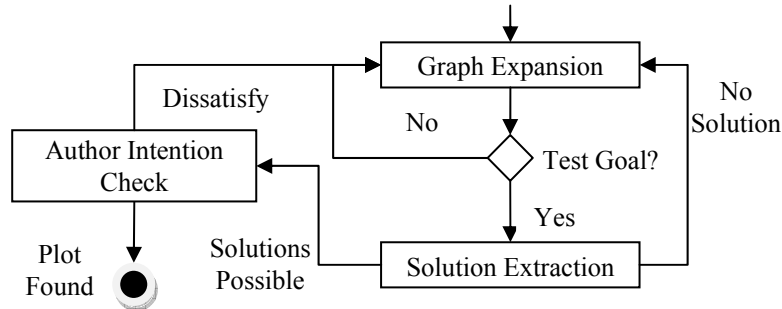


Fig. 1. A Framework of Plot Controllable Story Generation Model

In this paper, we describe a plot controllable story generation model that cleverly integrates author intentions into Graphplan based planner to generate more dramatic and interesting story plots. As illustrated in Fig. 1, the framework contains two phases: Graphplan phase and author intention check phase. In the Graphplan phase, the algorithm alternates execution of graph expansion and solution extraction until all solutions in current expansion level are found or it is proven that no solution exists. In the intention check phase, all possible solutions will be tested whether they satisfy author intentions or not. We conclude with a discussion of the limitations and future directions of this research. The main contribution of our work is that we use Graphplan as a computational story generation model and integrate author intentions into model to generate plot controllable stories.

2 Related Work

A majority of the works have looked to adapt the plan generation algorithm itself to structure the story along aesthetic lines. For example, to balance logical causal progression of plot and character believability, Riedl and Young describe a refinement search planning algorithm the IPOCL (Intent-based Partial Order Causal Link) planner - that, in addition to creating causally sound plot progression, reasons about character intentionality by identifying possible character goals that explain their actions and creating plan structures that explain why those characters commit to their goals [4]. Whilst Cheong use the Longbow planning algorithm to approximate the reader's planning related reasoning in order to demonstrate how the narrative structure can be post-processed to help promote suspense [5]. To produce conflict plot, Ware presents a CPOCL (Conflict Partial Order Causal Link) planner which allows narrative conflict to arise in a plan without destroying causal soundness [6]. To capture character's psychology, Pizzi uses HSP and its underlying real-time search algorithm to generate interactive affective plot according to the emotion of the story characters [7].

To make the generative plot under the control of the author, Riedl describe a general mechanism, called author goals, which can be used by human authors to assert authorial intent over generative narrative systems [8]. Similarly, Cavazza have developed

an approach to planning with trajectory constraints. The approach decomposes the problem into a set of smaller sub problems using the temporal orderings described by the constraints and then solves these sub problems incrementally [9]. Other researchers have proposed the use of search based drama management. The Carnegie Mellon University Oz project uses drama manager to prevent uninteresting and poorly structured stories from emerging [10]. A drama manager oversees and coordinates character agent behavior in order to coerce interesting and well-structured performances out of the autonomous agents.

3 Generating stories with controllable plot

3.1 Planning as a model of plot generation

There are many similarities between plan and story at the level of plot generation. In particular, a plot is a sequence of events that describes how the story world changes over time. In a plot, change is instigated by goal-directed actions of story world characters, although the story world can also be changed through unintentional acts such as accidents and forces of nature. Likewise, a plan is a set of ordered operators that transforms a world from one state to another state. If operators of a plan are events that can happen in a story world, then a plan can be a model of a plot generation.

Planners are implementations of algorithms that solve the planning problem: given a domain theory, an initial state I , and a goal situation G consisting of a set of propositions, and a sound sequence of actions that maps the initial state into a state where G is true. The domain theory is a model of how the world can change. For example, one can use STRIPS-like (Stanford Research Institute Problem Solver) operators that specify what operations can be performed in the world when they are applicable, and how the world is different afterwards. Various algorithms have been developed that solve planning problems including POP (Partial-Order Planners) [11], CSP (Constraint Satisfaction Planners) [12], Graphplan Planners [13] and HSP (Heuristic Search Planners) [14]. Since a plan can be used as a model of plot generation, a planning algorithm can also be used as a model of the dramatic authoring process to create plots. Thus, the creation of a plot can be considered a problem solving activity if one considers the plot of a story to be the sequence of story-world events that achieves some outcome desired by the author.

Graphplan is a planner for STRIPS domains. Planning in Graphplan is based on the concept of a data structure called the planning graph. It is considered a breakthrough in terms of efficiency regarding previous approaches to planning, and has been refined into a series of other, more powerful planners, such as IPP and STAN2, whose efficiency has been empirically verified in several planning algorithm competitions. The Graphplan algorithm performs a procedure close to iterative deepening, discovering a new part of the search space at each iteration. It iteratively expands the planning graph by one level, and then it searches backward from the last level of this graph for a solution. The iterative loop of graph expansion and search is pursued until either a plan is found or a failure termination condition is met.

3.2 Author intentions as constraints

In our computational representation of story plot, author intentions are implemented as a special type of plan steps that should occur in the intermediate world state. Author intentions can be used to force the planner to consider substantially more complex plans in which some intermediate steps, such as the character becomes poor or encounters robber, must be integrated into the resultant plans. Author intentions are provided at the time of planner initialization and describe world events or actions that must be achieved at some intermediate time during plan execution. If more than one author intention is given, there can be pre-specified temporal links between them so that author intentions must occur in a particular order.

To implement the ability for a planner to act on author intentions, we use state trajectory constraints, which is introduced in PDDL3.0 (Planning Domain Definition Language), as a reference [15]. State trajectory constraints assert conditions that must be met by the entire sequence of states visited during the execution of a plan. We recognize that there would be value in allowing propositions asserting the occurrence of action instances in a plan, rather than simply describing properties of the states visited during execution of the plan. The language provides a number of modal operators for expressing these constraints. In our computational model we used the basic modal operators: sometime, sometime-before, sometime-after, at-most-once and at-end. The sometime operator allows us to specify events that must occur in the story but not known when these events will happen. The sometime-before operator and sometime-after operator give a temporal order over these events. The at-most-once operator indicates that the events will happen not more than once. The at-end operator specifies the final events in the story. Fig. 2 shows an instance of author intentions.

```
(sometime (lose Tom Wedding-Ring))  
(sometime (rob Jack Tom Wedding-Ring))  
(sometime-before (lose Tom Wedding-Ring) (rob Jack Tom Wedding-Ring))  
(at-end (marry Tom Mary))  
(at-most-once (lose Tom Wedding-Ring))
```

Fig. 2. An Instance of Author Intentions

In this way, the existence of author intentions serves two important purposes. First, author intentions constrain the story search space such that it is impossible for a planner to produce a story that does not meet certain criteria imposed by the human author. That is, the planner cannot consider any plan in which the events described by an author intention will not be achieved during plan execution. Second, author intentions control plot generation to make the story more interesting and dramatic.

3.3 Integrating author intentions into plot planning

Although the Graphplan method has achieved good results in the field of traditional planning, it is also confronted with the problem we describe in the introduction section when applying Graphplan in story generation directly. To remedy this problem, we

extend the original Graphplan algorithm which integrates author intentions into planning. This method is called the plot controllable story generation method.

```

PC-Graphplan(A, s0, g, C)
1  i ← 0, ∇ ← ∅, P0 ← S0, G ← <P0>
2  until [g ⊆ Pi and g2 ∩ μPi = ∅] or Fixedpoint(G) do
3    i ← i+1
4    G ← Expand(G)
5  if g ⊄ Pi or g2 ∩ μPi ≠ ∅ then return(failure)
6  Π ← Extract (G, g, i)
7  if Fixedpoint(G) then η ← |∇(k)| else η ← 0
8  while Π=failure or not satisfy (Π, C) do
9    i ← i+1
10   G ← Expand(G)
11   Π ← Extract(G, g, i)
12   if Π=failure and Fixedpoint(G) then
13     if η=|∇(k)| then return failure
14   η ← |∇(k)|
15  return(Π)

```

Fig. 3. Plot Controllable Graphplan Algorithm

An outline of the plot controllable Graphplan algorithm is shown in Fig. 3. The input of the algorithm is $\langle A, s_0, g, C \rangle$ where: A is the union of all ground instances of operators in O ; s_0 and g are an initial state and goal condition of the world; and C , a set of constraints with author intentions. Upon successful termination of the algorithm, the output is a plan P that satisfies the constraints and achieves all final goals. The first step in the algorithm is the initial expansion of the graph until either it reaches a level containing all goal propositions without mutex or it arrives at a fixed-point level in G . If the latter happens first, then the goal is not achieved. Otherwise, a search for all solutions in current expansion level is performed. If no solution is found or all solutions are not satisfied constraint set C at this stage, the algorithm iteratively expands, and then searches the graph G . This iterative deepening is pursued even after a fix-point level has been reached, until success or the termination condition is satisfied. This termination condition requires that the number of no good tuples in $\nabla(k)$ at the fixed-point level k , stabilizes after two successive failures.

Evidently, the PC-Graphplan algorithm does not change the intrinsic complexity of planning, which is PSPACE-complete in the set-theoretic representation. Since the expansion of the planning graph and the author intention check are performed in polynomial time, this means that the costly part of the algorithm is in the search of the planning graph. Furthermore, the memory requirement of the planning-graph data structure can be a significant limiting factor.

4 Results

The central hypothesis of our work is that author intentions can be used for controlling plot in the story generation process. In this section we present the results of a qualitative evaluation that support this hypothesis, via analysis of a selection of sample plot. For the evaluation we developed a Graphplan style planner which is an implementation of the algorithm outlined in fig. 2. In these experiments performance was found to be acceptable for story generation purposes. Our results are encouraging and support the hypothesis that these constraints can be used for controlling the plot generation.

```
(define (problem marry-a-girl)
  :inits (and ((single Tom) (single Mary) (loves Tom Mary) (has Tom Money)))
  :author_goal (married Tom Mary)
  :author_intention (sometime (lose Tom Wedding-Ring)))
```

Fig. 4. Planning Problem Definition in “marry a girl”

The uses of author intentions in the story marry a girl illustrate their necessity in controlling the story plot. Fig. 4 shows the problem initialization in a PDDL-like language. An initial state defines characters, character traits, and relevant props and features of the world. The outcome is the author goal: Tom married Mary. The planning system is also initialized with an action library that describes ways in which the world can change. For example, characters can buy things if they have money and one character can marry another character if they love each other.

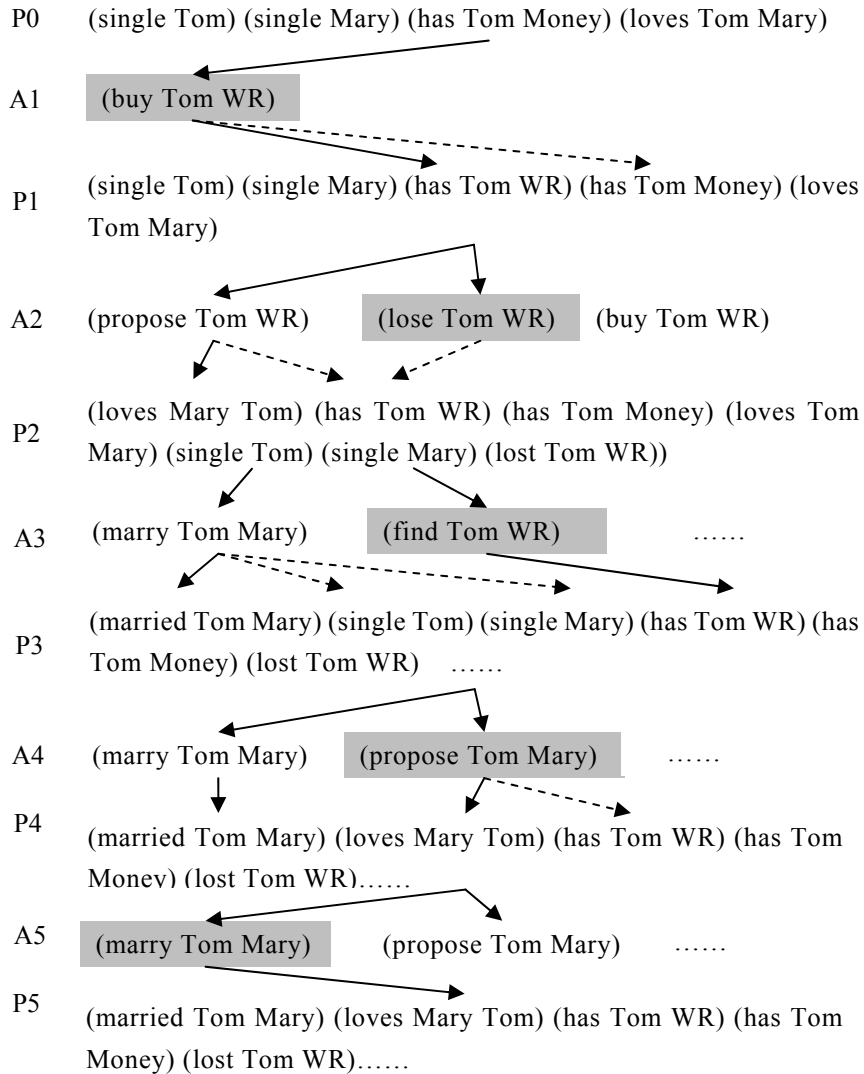


Fig. 5. Planning Result in “marry a girl”

The author intention defines that a significant feature of generation is that Tom will lose the wedding ring at some point. Note that the initialization parameters do not indicate how the author intention or the outcome is achieved, only that they must be achieved. Author intentions are necessary for controlling the plot. Without author intention, a planning algorithm could naively generate the following: Tom buys wedding ring. Tom proposes Mary. Tom marries with Mary. The author intention prevents this by forcing the planner to consider substantially more complex plans in which Tom loses the wedding ring. Fig. 5 shows one possible plot that can be generated by respecting the author intention. A_i is the set of actions whose preconditions are nodes in

P_{i-1} . P_i is defined as the union of P_{i-1} and the sets of positive effects of actions in A_i . Solid arrows are preconditions of actions or positive effects of actions. Dashed arrows are negative effects of actions. The result plot is: Tom buys wedding ring. Tom loses the wedding ring. Tom finds the wedding ring. Tom proposes Marry. Tom marries with Mary.

5 Conclusions and Future Work

The author intentions described in this paper is an attempt to enable human authors to inject their individual preference and requirements into the story generation. In general, author intentions constrain the planner to produce plans with particular structures by pruning branches of the plan search space in which plans do not meet the author intentions. Aside from controlling plot in the story generation, Author intentions also have the pragmatic side effect that they can generate more complex plan solutions. We believe that enabling the author to inject their intentions into the planning process will become increasingly important to generate controllable and dramatic story plots.

Although we have investigated several important aspects of plot controllable story generation model, there are still some of problems worth further consideration. 1) Classify author intentions in more detail. Different type of author intentions will generate different dramatic plots. For example, some intentions will generate conflict plots and others will generate suspense plots and etc. For the purpose of produce the plots we need, the explicit representation and classification of author intentions in story generation are very necessary. 2) From static author intentions to dynamic author intentions. In this work, the author intentions are predefined before plot planning. Although they can effective control of the development of plot, they cannot revise the intentions in the process of plot planning. In the future, we will design intention revision rules to make the plot planner modify author intentions autonomously and dynamically.

Acknowledgments

This work is supported by the National Natural Science Foundation of China under grant No.61035004, 61173063, 30973716 and National Social Science Foundation of China under grant No.10AYY003.

References

1. Ruqian Lu., Songmao Zhang.: Automatic Generation of Computer Animation: Using AI for Movie Animation, Berlin, Heidelberg: Springer-Verlag, 2002.
2. Cungen Cao et al.: Progress In The Development of Intelligent Animation Production, Journal of Systems Science and Mathematical Sciences, vol. 28 no. 11, 2008, p.1407-1431.
3. Haitao Wang.: Research on Text Knowledge Processing and its Application in Intelligent Narrative Generation, Chinese Academy of Sciences, 2008.

4. Riedl M.O., Young R.M.: Narrative Planning: Balancing Plot and Character, *Journal of Artificial Intelligence Research*, vol 39: 217-268, 2010.
5. Cheong G.Y.: A Computational Model of Narrative Generation for Suspense, North Carolina University, 2007.
6. Ware S.G., Young R.M.: Modelling Narrative Conflict to Generate Interesting Stories, In *Proceedings of Artificial Intelligence in Interactive Digital Entertainment, AIIDE*, 2010.
7. Pizzi D.: Emotional Planning for Character-based Interactive Storytelling, Teesside University, 2011.
8. Riedl M.O.: Incorporating authorial intent into generative narrative systems, *Intelligent Narrative Technologies II: Papers from the 2009 Spring Symposium (Technical Report SS-09-06)*, pp. 91-94, Palo Alto, CA. AAAI Press. 2009.
9. Porteous J., Cavazza M.: Controlling narrative generation with planning trajectories: the role of constraints, In *Proceedings of the 2nd International Conference on Interactive Digital Storytelling*, pp. 234-245. 2009.
10. Kelso M.T., Weyhrauch P., Bates J.: Dramatic Presence, *The Journal of Tele operators and Virtual Environments*, vol. 2 no. 1, p. 1-15, 1993.
11. Penberthy J.S., Weld, D.: UCPOP: A Sound, Complete, Partial Order Planner for ADL, In *Proc. KR-92*, 1992.
12. Do M., Kambhampati S.: Planning as constraint satisfaction: solving the planning graph by compiling it into CSP, *Artificial Intelligence*, vol. 132 no. 2, 2001, p.151– 182.
13. Blum A., Furst M.: Fast planning through planning graph analysis, In *Proc. IJCAI*, 1995.
14. Blai B., Héctor G.: Planning as heuristic search, *Artificial Intelligence*, 2001, P. 5-33.
15. Edelkamp S., Jabbar S., Nazih M.: Large-Scale Optimal PDDL3 Planning with MIPS-XXL, In *Proceedings of Planning Competition, ICAPS*, 2006.