

Glaive: A State-Space Narrative Planner Supporting Intentionality and Conflict

Stephen G. Ware

sgware@uno.edu

Computer Science Department
University of New Orleans
New Orleans, LA, USA

R. Michael Young

young@csc.ncsu.edu

Liquid Narrative Group, Computer Science Dept.
North Carolina State University
Raleigh, NC, USA

Abstract

Glaive is a state-space planner based on Hoffmann and Nebel's Fast-Forward (2011) which solves the narrative planning problem defined by Riedl and Young (2010)—to construct a plan which achieves the author's goals out of steps which are clearly motivated and goal-oriented toward individual character goals. Glaive reasons about how characters cooperate and conflict based on causal structures and possible worlds. By leveraging the unique constraints of narrative planning, Glaive reduces its branching factor and calculates a more accurate heuristic. We evaluate it on 8 narrative planning problems and demonstrate that it can solve certain non-trivial problems in under 1 second.

Introduction

AI planning has proven a popular paradigm for developing computational models of narrative and interactive narrative systems (Young 1999; Young et al. 2014). Plan-based models are attractive because they can be generated and adapted by planning algorithms; however, most planning-based narrative systems have focused either on a rich knowledge representation or on the use of fast planning algorithms, often at the expense of the other.

For example, numerous narrative phenomena have been modeled by extending the Partial Order Causal Link (POCL) planning paradigm: character intentionality (Riedl and Young 2010), suspense (Bae and Young 2008), and conflict (Ware and Young 2011), to name a few. POCL plans are rich data structures with explicit representations of causal and temporal structure that facilitate reasoning about other aspects of narrative. Unfortunately, POCL algorithms are often too slow to be used in an interactive context, even for small problems, and these extensions to the plan representation slow the process even further.

Alternatively, fast planning algorithms have been used at run time to create interactive stories, including experiences based on the TV show *Friends* (Cavazza, Charles, and Mead 2002), Flaubert's novel *Madame Bovary* (Pizzi and Cavazza 2007), and Shakespeare's play *The Merchant of Venice* (Porteous, Cavazza, and Charles 2010), to name a few. These

systems tend not to modify the plan representation or planning algorithm, but instead encode narrative constraints directly into the story domain or as control knowledge on a per-story basis. This is reasonable given that each story domain has its own structure, but some narrative phenomena—like intentionality and conflict—are universal enough to be reasoned about directly by the planner. This has the potential to reduce the knowledge engineering burden and increase the efficiency of the planner by avoiding non-narrative plans in the search space.

Glaive is a state-space heuristic search planning algorithm which reasons directly about character intentionality and represents alternate worlds to facilitate reasoning about phenomena like conflict. This paper describes the Glaive algorithm, its heuristic, and the evaluation of Glaive on a set of benchmark problems.

Related Work

Glaive solves the intentional planning problem described by Riedl and Young (2010): A valid plan is one which achieves the author's goals but is only composed of steps that are explained in terms of the individual goals of the characters who take them. This problem was extended by Ware and Young (2011) to include solutions in which the plans of some characters fail. Narratologists have described conflict in terms of the thwarted plans of intentional agents (Herman, Jahn, and Ryan 2005), so failed plans are an important aspect of both static and interactive narratives. These models of intentionality and conflict are described in terms of the causal structure of the story, which human audiences reason about when experiencing a narrative (Trabasso and Van Den Broek 1985).

Narrative generating systems are often divided into the *strong story* or *strong autonomy* camps (Riedl and Bulitko 2013). The former ensures a unified plot defined by the author, whereas the later ensures an accurate simulation of each character. Intentional planning is a compromise between the two—it ensures the author's desired outcome while generating believable character behavior. The framework can represent cooperation between agents (steps which contribute to more than one character's goals) and accidents (steps which contribute to no character's goals). This kind of coordination is difficult when each character is controlled by a separate planning process, which is usually the case in

strong autonomy systems, but can be accomplished (Teutenberg and Porteous 2013). Using a single planner makes coordination easier but is hampered by the intractability of planning (Helmert 2006b).

In summary, the problem Glaive solves is a multi-agent coordination problem in which characters sometimes cooperate and sometimes conflict as they are guided by an invisible puppet master toward the author’s goal. This framework lends itself nicely to narrative domains.

The IPOCL (Riedl and Young 2010) and CPOCL (Ware and Young 2011) planners solve the narrative planning problem, but the speed of these plan-space search algorithms has prevented their use at run time in interactive experiences. Forward-chaining state-space heuristic search planners have emerged as the dominant technology in the biannual International Planning Competition due to their speed (Coles et al. 2011). Glaive is based on Hoffmann and Nebel’s Fast-Forward planner (Hoffmann and Nebel 2011), but like the IPOCL and CPOCL algorithms, Glaive also tracks the causal history of each proposition. Glaive is a hybrid of these two planner families; it attempts to combine the speed of Fast-Forward with the causal reasoning capabilities of IPOCL and CPOCL.

Narrative Planning

An intentional planning domain defines parametrized actions that can occur as events in a story. An action has preconditions which must be true before it occurs, effects which become true after it occurs, and a (possibly empty) set of characters who must consent to take that action.

An intentional planning problem defines the initial state of the story world and a set of author goals which must be true by the time the story has finished. The solution to such a problem is a plan, which is a sequence of fully ground actions called steps.

In addition to the author’s goals, Glaive tracks individual goals for each character which may be adopted and abandoned at various times during the story. The *c intends g* modality is used to indicate that character *c* has adopted goal *g* and may now take actions to make *g* true.

Figure 1 gives an example domain and problem which models a highly simplified version of the film *Indiana Jones and the Raiders of the Lost Ark*. There are 4 types of actions:

1. *dig*: A character discovers and excavates a buried item.
2. *open*: A character opens the Ark and dies from it.
3. *give*: One character gives an item to another.
4. *take*: One character takes an item from another. Either the other is dead or the thief takes the item at gunpoint.

Initially, all three characters are alive. The Ark is buried but Indiana Jones knows where to find it. Both Indiana and the US Army intend that the army should have the Ark. The Nazis are armed, and they intend to open the Ark (but they don’t know where to find it). The author’s goals for the end of the story are that the US Army should have the Ark and the Nazis should be dead.

Figure 2 gives the solution: First Indiana excavates the Ark. He intends to give it to the US Army, but before he

can do that the Nazis take it from him at gunpoint. The Nazis then open the Ark and die. Finally, the US Army takes the Ark from the dead antagonists.

Intentional Paths

A Glaive plan explicitly tracks how earlier steps satisfy the preconditions of later steps.

Definition 1. A *causal link* $s \xrightarrow{p} t$ exists from step *s* to step *t* for proposition *p* if and only if step *s* has effect *p*, step *t* has precondition *p*, and none of the steps that occur between *s* and *t* have effect $\neg p$. We say *s* is the causal parent of *t*, and that a step’s causal ancestors are the steps in the transitive closure of this relationship.

A plan represents the entire story, but it contains subsequences of steps which correspond to the plans of each individual character. These character plans are described in terms of character goals and causal structures.

Definition 2. A *intentional path* for some character *c* and some goal *g* is an alternating sequence of *n* steps and *n* propositions $\langle s_1, p_1, s_2, p_2, \dots, s_n, g \rangle$ such that:

1. Character *c* must consent to all steps.
2. *c intends g* is true before s_1 and true until step s_n .
3. Step s_n has effect *g*.
4. For *i* from 1 to $n - 1$, there exists a causal link $s_i \xrightarrow{p_i} s_{i+1}$.
5. No proposition appears twice.
6. The path never contains a proposition and its negation.

An intentional path describes a sequence of steps taken by a character in service of a goal. Consider this intentional path:

$$\langle \text{dig}(J, R), J \text{ has } R, \text{give}(J, R, U), U \text{ has } R \rangle$$

Indiana excavated the Ark so he could possess the Ark so he could give it to the US Army, which achieves his goal that the army have the Ark.

Note that there can be 0, 1, or many intentional paths for each character goal. These paths may overlap. Glaive reasons about intentional paths when calculating its heuristic. Intentional paths also allow us to define which plans are solutions to the narrative planning problem.

Definition 3. A step *s* is *explained* if and only if:

1. \forall consenting character *c*, *s* is on an intentional path for *c*.
2. All other steps on that intentional path are explained.

In other words, every character who takes *s* has a reason to take *s*, and the other steps used to explain *s* are also reasonable steps with explanations of their own.

Definition 4. A *valid intentional plan* is a sequence of steps such that:

1. Each step’s preconditions are satisfied immediately before the step is taken.
2. After all steps are taken, the author’s goals are satisfied.
3. Every step is explained in some possible world.

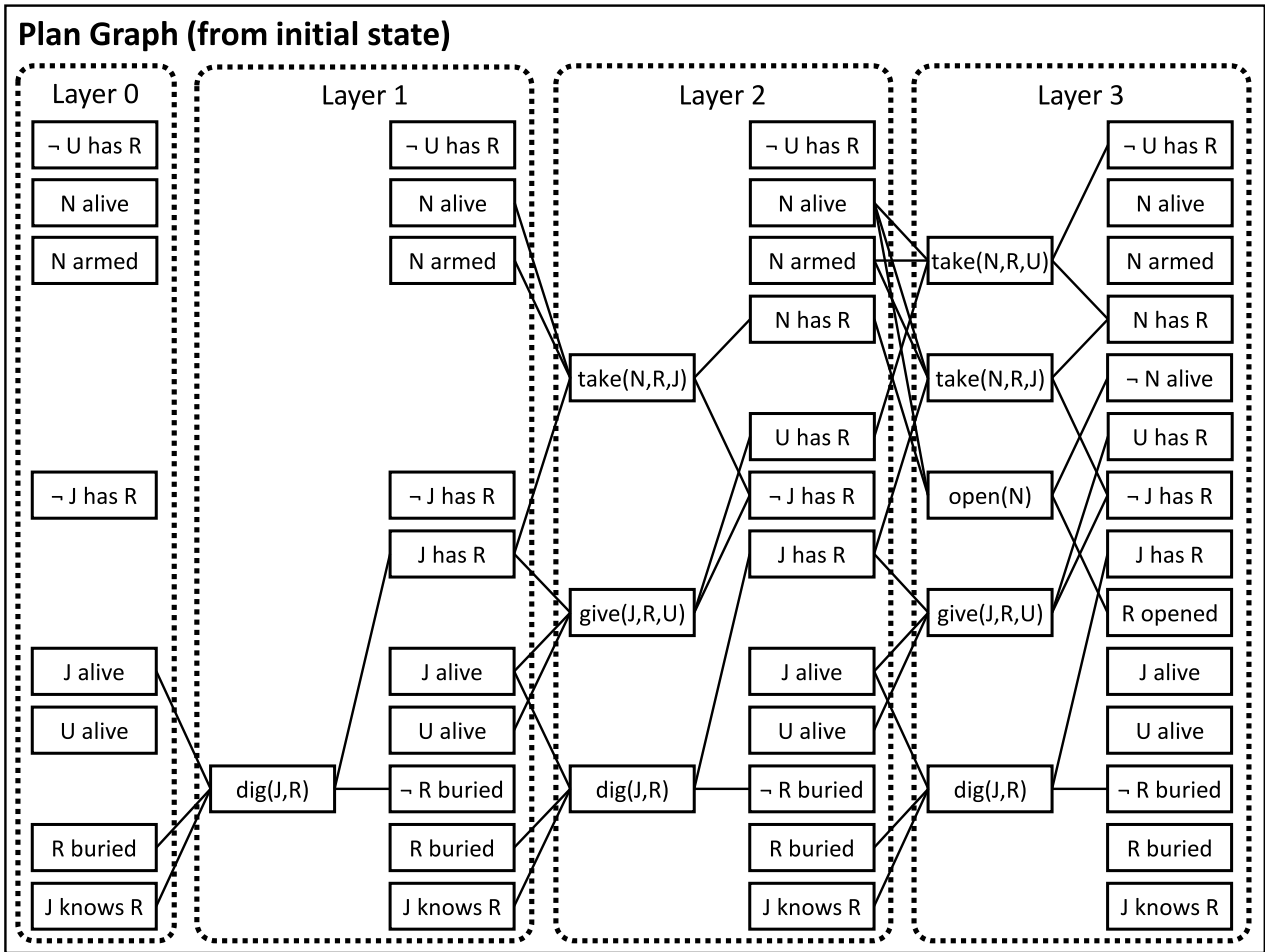
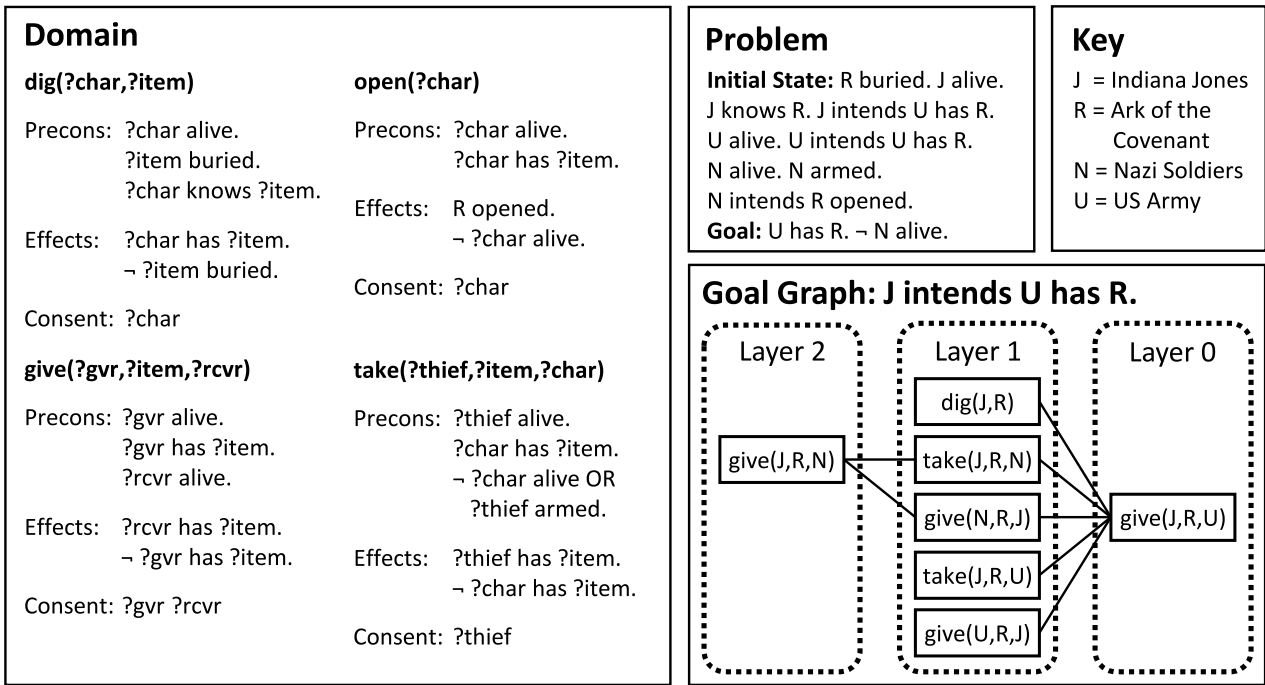


Figure 1: A highly simplified example of the plot for *Indiana Jones and the Raiders of the Lost Ark*.

Algorithm 1 GLAIVE(Π, σ, G, U)

```
1: Let  $\Pi$  be the plan,  $\sigma$  the current state,  $G$  the set of character goals, and  $X$  the set of unexplained steps.
2: Nondeterministically choose a potentially motivated step  $s$  whose preconditions are satisfied in  $\sigma$ .
3: Add step  $s$  to  $\Pi$ .
4: Apply the effects of  $s$  to  $\sigma$ .
5: for each effect of  $s$  like  $c$  intends  $g$  do
6:   Add a new character goal  $\langle c, g \rangle$  to  $G$ .
7: end for
8: if any characters consent to  $s$  then
9:   Add  $s$  to  $X$ .
10: end if
11: for each character goal  $g = \langle c, g \rangle \in G$  do
12:   for each intentional path  $p$  for  $c$  ending in  $g$  do
13:     Remove  $g$  from  $G$ .
14:     for each step  $t \in p$  do
15:       if  $t$  is explained then
16:         Remove  $t$  from  $X$  for all nodes*.
17:       end if
18:     end for
19:   end for
20: end for
21: if any node* satisfies the author's goals and  $X = \emptyset$  then
22:   return  $\Pi$  for that node.
23: else
24:   GLAIVE( $\Pi, \sigma, G, X$ )
25: end if
```

This idea of a step being explained *in some possible world* means that the step is part of some character's plan even if that plan fails. This will be discussed in more detail below, but that discussion first requires an explanation of how Glaive performs its search.

The Glaive Algorithm

Glaive is a state-space planner, meaning that it begins at the initial state of the problem and takes steps which change that state until it finds a state in which the author's goals are true. Its search space can be envisioned as a directed tree (see Figure 2). A node in the tree represents a state; an edge $n_1 \xrightarrow{s} n_2$ represents applying the effects of step s to the state of node n_1 to produce the new node n_2 with a different state. In practice, a node also represents a plan made of the steps taken on the path from the root to that node. The root of the tree is the initial state of the problem and an empty plan.

Algorithm 1 describes how Glaive performs its search. In addition to the current plan and current state, Glaive tracks two additional things: a set of character goals G and a set of unexplained steps X .

Definition 5. A *character goal* is a 2-tuple $\langle c, g \rangle$ which represents that character c intends goal g in the current state.

When Glaive takes a step which has an effect like c intends g , it adds a new character goal to G (line 6). Once an intentional path is found for character c that ends in g , that goal is removed (line 13).

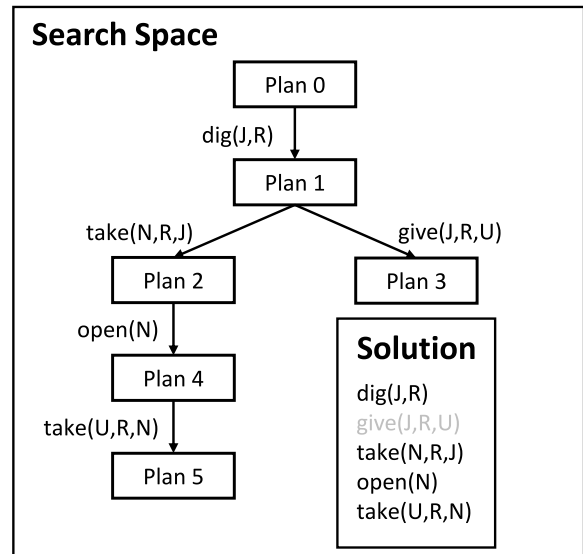


Figure 2: A search space and solution for the example problem in Figure 1. The step in gray indicates a step that Indiana intended to take but did not take due to a conflict.

An unexplained step implies a commitment to explain why the characters took that step. When a step s with one or more consenting characters is taken, it gets added to X (line 9). When an intentional path is found that contains step s , we check to see if s is explained, and if so remove it from X (line 16).

Glaive returns a solution if it finds a node where the author's goals are satisfied and no steps remain unexplained. The plan to reach such a node will be a valid plan.

Possible Worlds and Conflict

A character's plans may fail during a narrative; indeed, this is a key element of conflict (Herman, Jahn, and Ryan 2005). Glaive reasons about failed plans by treating a state space not just as a data structure for performing its search but also as a representation of many possible worlds.

Consider line 13 in Algorithm 1 (marked with an *). When a step becomes explained, it is removed from X not only for the current node but also for every node in the search space where that instance of that step is unexplained.

The search space in Figure 2 provides an example. It begins in the initial state with Plan 0 as an empty plan. The first step taken is $dig(J, R)$, which produces Plan 1. This step is unexplained because it does not directly satisfy any of Indiana's goals, so it is in the set X . When the right branch of the space is expanded by taking $give(J, R, U)$ as the second step and creating Plan 3, this intentional path is formed:

$$\langle dig(J, R), J \text{ has } R, give(J, R, U), U \text{ has } R \rangle$$

It provides an explanation for both of the steps on that path. The step $give(J, R, U)$ is removed from X in Plan 3. The step $dig(J, R, U)$ is removed from X in Plan 3 and also in Plan 1 and all of its descendants. This is because there exists a possible world in which it makes sense for Indiana

to excavate the Ark of the Covenant—that world is Plan 3. However, Plan 3 does not satisfy the author’s goal that the Nazis be dead, so it is not a solution.

Now consider the left branch of the search space. Plan 5 satisfies all of the author’s goals, but the very first step in Plan 5, $dig(J, R)$, is not on any intentional path in Plan 5. However, that step is not unexplained. It was explained by Plan 3, even though Plan 3 is not an ancestor of Plan 5.

In other words, once a character plan is formed in some possible world (such as Indiana’s 2 step plan to get the Ark to the US army), any prefix of that plan can appear in other possible worlds and still be explained.

Note that Plan 5 does not become a solution until Plan 3 is discovered. This is why line 21 (marked with an *) states that any node can be returned as a solution even if it is not the current node. Even if Plan 5 is discovered before Plan 3, Plan 5 is still returned as a solution once all of its steps become explained—that is, once Plan 3 is discovered.

Representing possible worlds allows Glaive to reason about how a character’s plans fail. Before returning a solution, Glaive can combine multiple nodes into a single solution to explain each character’s actions. The example solution given in Figure 2 is a combination of Plan 5 and Plan 3. The step which appears in Plan 3 but not in Plan 5 is shown in gray to indicate that it is a non-executed step. In other words, it does not actually happen in the story, but it tells us what Indiana was planning and why he excavated the Ark. This use of non-executed steps to represent thwarted plans mirrors Ware and Young’s threatened causal link representation of narrative conflict (2011), and it has been demonstrated that human audiences recognize these kinds of thwarted plans when reading stories (Ware and Young 2012).

The Glaive Heuristic

Like other state-space planners, the Glaive heuristic is responsible for the planner’s speed. A heuristic is a function $h(n)$ which, given some node n in the search space, estimates how many more steps need to be taken before a solution is discovered.

The Glaive heuristic is calculated as the maximum of two numbers: an estimate derived by reasoning backward from each character goal and an estimate derived by reasoning forward from the current state to the author’s goals. Glaive uses two kinds of graphs to calculate these numbers: goal graphs and plan graphs respectively.

Goal Graphs

Definition 6. A goal graph is a directed, layered graph composed of steps. It is constructed for some character c and some goal g . A step s exists at layer 0 iff c consents to s and s has g as an effect. A step s exists at layer $i \geq 0$ iff c consents to s , s does not exist at any earlier layer, and there exists a proposition p such that s has effect p and some step at layer $i - 1$ has precondition p . An edge $s_1 \xrightarrow{p} s_2$ exists from s_1 to s_2 iff s_1 exists at layer i and has effect p , and s_2 exists at layer $i - 1$ and has precondition p .

An example goal graph is given in Figure 1. It contains all the steps which Indiana might possibly consent to while

pursuing his goal that the US Army have the Ark. Only the step $give(J, R, U)$ directly achieves this goal, so only it appears at Layer 0. All the steps that require his consent and which can satisfy some precondition of that step appear at Layer 1, and so on. Note that $give(J, R, U)$ does not appear at Layer 2 because it already exists at Layer 0 and cannot be repeated. Also note that goal graphs do not change based on the current state, so they only need to be computed once during the search process.

Goal graphs allow Glaive to reduce the branching factor of its search space. When choosing a next step (line 2), Glaive only considers those steps whose preconditions are satisfied and which might eventually be explained.

Definition 7. A step s which requires the consent of character c is *potentially motivated for c* iff there exists some goal g such that c *intends* g is true in the current state and s appears somewhere in the goal graph for c *intends* g . A step is *potentially motivated* (in general) if it is potentially motivated for all its consenting characters.

A step which is not potentially motivated can never be explained in the future, so there is no reason to consider it as a next step during the search.

A goal graph represents all the possible intentional paths that might exist for character c to achieve goal g . When some step is unexplained, Glaive uses the layer at which that step appears in a goal graph to estimate how difficult it will be to explain it. Consider the cost of explaining $dig(J, R)$ in Plan 1. It is potentially motivated by the goal J *intends* U *has* R because it appears in the goal graph for that goal. Because it appears at layer 1, we know that at least 1 more step is required before that explanation can be used.

It is possible that multiple unexplained steps will eventually be explained by a single step—that is, their intentional paths will overlap. To avoid overestimating, Glaive considers this.

Definition 8. A step s in a goal graph is *dominated* by a step t iff there exists a goal graph for some current character goal containing s and t which has an edge $s \rightarrow t$.

Now we can define how Glaive estimates the number of additional steps needed to explain an unexplained step. Let x be some unexplained step. Let C be the set of characters which must consent to s . Let $\Gamma(x, c)$ be all the goal graphs for c in which x appears and for which there exists a goal $\langle c, g \rangle \in G$. Let the function $layer(x, \gamma)$ denote the layer at which x appears in goal graph γ .

$$\text{cost}(x) = \sum_{c \in C} \left(\min_{\gamma \in \Gamma(x, c)} \begin{cases} 0 & \text{if } x \text{ is dominated in } \gamma \\ layer(x, \gamma) & \text{otherwise} \end{cases} \right)$$

This cost is calculated for every unexplained step and the total is used as part of Glaive’s heuristic. For some node n in the search space, let $\text{cost}(n)$ be the sum of the cost of all unexplained steps $x \in X$.

Plan Graphs Glaive’s plan graphs are an extension of those used by Hoffmann and Nebel’s Fast-Forward (2011). Plan graphs have layers which contain propositions and steps. The first layer has those propositions which are true

Table 1: Performances for the Fast-Forward and Glaive heuristics on 8 benchmark narrative planning problems.

Problem	Planner	Sol?	Time	Visited	Expanded
Space	FF	Yes	0.00	3	15
	Glaive	Yes	0.00	3	9
Fantasy	FF	Yes	2.76	35,407	212,046
	Glaive	Yes	0.03	14	107
Raiders	FF	Yes	0.23	1,334	4,073
	Glaive	Yes	0.03	35	142
Aladdin	FF	No	288.49	67,758	2,063,927
	Glaive	Yes	0.06	12	189
BLP-Win	FF	Yes	27.38	32,262	196,576
	Glaive	Yes	0.31	109	586
Western	FF	No	440.55	166,110	3,079,097
	Glaive	Yes	28.41	18,855	296,150
BLP-Die	FF	No	195.93	106,750	731,948
	Glaive	No	236.97	91,887	720,024
Heist	FF	No	367.92	84,854	1,654,916
	Glaive	No	351.38	115,782	1,544,048

in the current state, and the propositions and steps at each layer increase monotonically. A plan graph represents a relaxed version of a planning problem where the delete lists of the steps are ignored. An example plan graph is given in Figure 1. The plan graph is extended until all of the author’s goals appear on the same layer, and then a solution to this relaxed problem is extracted. The length of that relaxed solution is how Fast-Forward estimates the number of steps remaining before the goal is achieved. Let that estimate be denoted as $FF(n)$ for some node n in the search space.

Due to space constraints, formal definitions cannot be provided here. The only change made to Glaive’s plan graphs is that a step may not appear at a layer until it is potentially motivated in the previous layer. Similarly, when a step gets included in a relaxed solution, one of its motivations must also be included. See Hoffmann and Nebel’s article (2011) for details on this process.

Glaive’s heuristic considers estimates derived from the goal graphs and plan graph. Let n be some node in the search space:

$$h(n) = \max(FF(n), \text{cost}(n))$$

The maximum of these two estimates is used (rather than the sum) because they are likely to consider some of the same steps and Glaive attempts to avoid overestimating. Glaive’s heuristic could be improved if double counting could be efficiently avoided.

Evaluation

Given the complexity of planning and the scope of most planning problems, a planner is usually evaluated on a suite of benchmark problems. We compiled 8 narrative planning problems to evaluate Glaive. Below is a brief explanation of where each problem came from and its size, given as *# of literals / # of steps / # of axioms / length of shortest solution*. These sizes are reported after the problems had been algorithmically simplified by Glaive as a pre-processing step.

1. *Space* (46 / 23 / 0 / 2), from Ware and Young (2012).

2. *Fantasy* (80 / 46 / 0 / 6), from Ware and Young (2012).
3. *Raiders* (46 / 68 / 6 / 8), a longer version of Figure 1.
4. *Aladdin* (294 / 213 / 165 / 11), from Riedl and Young (2010).
5. *BLP-Win* (215 / 705 / 632 / 10), simplest way to win the interactive narrative game *The Best Laid Plans* (Ware et al. 2014).
6. *Western* (67 / 632 / 0 / 7), from Ware and Young (2012).
7. *BLP-Win* (215 / 705 / 632 / 11), fastest way to die in *The Best Laid Plans* (Ware et al. 2014).
8. *Heist* (323 / 1844 / 0 / 31), from Niehaus (2009).

Table 1 gives Glaive’s performance on these problems. The planner uses complete A* search and was given 6 Gb of memory on a computer with a 3.5 GHz Intel Core I7 processor. As a basis for comparison, we also show how Glaive performs when using only the Fast-Forward heuristic instead of Glaive’s heuristic. Time is given in seconds as the average of 10 runs.

In all cases Glaive performs comparably or better when using its heuristic, often significantly better. Riedl and Young (2010) report that the original IPOCL planner took over 12 hours to solve *Aladdin*, visited 673,079 nodes and expanded 1,857,373 while using a domain-specific heuristic. By contrast, Glaive uses a domain independent heuristic, takes only 64 milliseconds, visits 12 nodes and expands 189. Given that Glaive’s solution is only 11 steps long, Glaive visits only 1 node that is not on the direct path to the solution.

Software

Glaive has been implemented in Java 7. The planner, along with the benchmark problems tested above, can be downloaded from:

<http://stephengware.com/projects/glaive/>

Glaive takes as input domains and problems in the Planning Domain Definition Language, a standard in the AI planning community. It supports a number of helpful planning features including typed constants, equality, disjunctive goals, universal and existential quantifiers, conditional effects, domain axioms, and the ability for one character to delegate its goals to others.

Future Work

Glaive was built on top of Fast-Forward due to its simplicity, speed, and use of data structures needed by Glaive. However, the Fast-Downward planner (Helmert 2006a) has proven itself faster, more accurate, and more memory efficient. More importantly, the Fast-Downward heuristic decomposes problems into causal sequences, which should facilitate reasoning about intentional paths. Glaive leverages the constraints of narrative problems to reduce its branching factor and increase the accuracy of the Fast-Forward heuristic. We believe that another order of magnitude speedup can be achieved by doing the same with Fast-Downward.

We are also excited to explore what other narrative phenomena besides conflict can be reasoned about using

Glaive's possible worlds representation. Narratologists have analyzed stories in terms of possible worlds (Ryan 1991), and there are established logical formalisms for possible worlds (e.g. Kripke (1963)) which can be used to develop computational models of narrative.

Conclusion

Glaive is a state-space narrative planner. It constructs stories which achieve the author's goals out of steps which are clearly motivated and goal-oriented for the characters who take them. It reasons about how characters cooperate when a step has multiple consenting characters. It also reasons about failed plans and conflict by treating its search space as a set of possible worlds. By leveraging the constraints of narrative planning, Glaive can reduce its branching factor and calculate a more accurate heuristic. Glaive is fast enough solve certain non-trivial problems fast enough for use at run time in interactive experiences such as the forthcoming narrative adventure game *The Best Laid Plans* (Ware et al. 2014).

Acknowledgements

This research was supported by NSF award IIS-0915598.

References

- Bae, B., and Young, R. 2008. A use of flashback and foreshadowing for surprise arousal in narrative using a plan-based approach. In *Proceedings of the International Conference on Interactive Digital Storytelling*, 156–167. Springer.
- Cavazza, M.; Charles, F.; and Mead, S. J. 2002. Character-based interactive storytelling. *Intelligent Systems* 17–24.
- Coles, A. J.; Coles, A. I.; Garca Olaya, A.; Jimnez, S.; Linares Lopez, C.; Sanner, S.; and Yoon, S. 2011. A survey of the seventh international planning competition. *AI Magazine*.
- Helmert, M. 2006a. The fast downward planning system. *Journal of Artificial Intelligence Research* 26(1):191–246.
- Helmert, M. 2006b. New complexity results for classical planning benchmarks. In *Proceedings of the International Conference on Automated Planning and Scheduling*, 52–61.
- Herman, D.; Jahn, M.; and Ryan, M. L. 2005. *Routledge encyclopedia of narrative theory*. Routledge.
- Hoffmann, J., and Nebel, B. 2011. The ff planning system: Fast plan generation through heuristic search. *arXiv preprint arXiv:1106.0675*.
- Kripke, S. A. 1963. Semantical considerations on modal logic. *Acta Philosophica Fennica* 16:83–94.
- Niehaus, J. M. 2009. *Cognitive models of discourse comprehension for narrative generation*. Ph.D. Dissertation, North Carolina State University.
- Pizzi, D., and Cavazza, M. 2007. Affective storytelling based on characters feelings. In *Proceedings of the AAAI Fall Symposium on Intelligent Narrative Technologies*, 111–118.
- Porteous, J.; Cavazza, M.; and Charles, F. 2010. Applying planning to interactive storytelling: Narrative control using state constraints. *ACM Transactions on Intelligent Systems and Technology (TIST)* 1(2):10.
- Riedl, M. O., and Bulitko, V. 2013. Interactive narrative: An intelligent systems approach. *AI Magazine* 34(1).
- Riedl, M. O., and Young, R. M. 2010. Narrative planning: balancing plot and character. *Journal of Artificial Intelligence Research* 39:217–268.
- Ryan, M. L. 1991. *Possible worlds, artificial intelligence, and narrative theory*. Indiana University.
- Teutenberg, J., and Porteous, J. 2013. Efficient intent-based narrative generation using multiple planning agents. In *Proceedings of the 2013 International Conference on Autonomous Agents and Multi-Agent Systems*, 603–610.
- Trabasso, T., and Van Den Broek, P. 1985. Causal thinking and the representation of narrative events. *Journal of memory and language* 24(5):612–630.
- Ware, S. G., and Young, R. M. 2011. CPOCL: A Narrative Planner Supporting Conflict. In *Proceedings of the 7th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*.
- Ware, S. G., and Young, R. M. 2012. Validating a Plan-Based Model of Narrative Conflict. In *Proceedings of the 7th International Conference on the Foundations of Digital Games*, 220–227.
- Ware, S. G.; Young, R. M.; Stith, C.; and Wright, P. 2014. *The Best Laid Plans*. <http://liquidnarrative.csc.ncsu.edu/blp/>. [Accessed August 11, 2014].
- Young, R. M.; Ware, S. G.; Cassell, B.; and Robertson, J. 2014. Plans and planning in narrative generation: A review of plan-based approaches to the generation of story, discourse and interactivity in narratives. *SDV. Sprache und Datenverarbeitung*. (forthcoming).
- Young, R. M. 1999. Notes on the use of plan structures in the creation of interactive plot. In *Proceedings of the Association for the Advancement of Artificial Intelligence Fall Symposium on Narrative Intelligence*, 164–167.