Writer's Aid: Using a Planner in a Collaborative Interface

Tamara Babaian
CIS Department, Bentley College,
Waltham, MA 02452
tbabaian@bentley.edu

Barbara J. Grosz
DEAS, Harvard University
Cambridge, MA 02138
grosz@deas.harvard.edu

Stuart M. Shieber
DEAS, Harvard University
Cambridge, MA 02138
shieber@deas.harvard.edu

Abstract

Writer's Aid is a collaborative system that works simultaneously with an author editing a document providing assistance in identifying and inserting citation keys and autonomously finding and caching potentially relevant papers and associated bibliographic information from various on-line sources. At the core of Writer's Aid is a domain-independent planning system. The use of a planner in Writer's Aid enables an efficient division of labor between a human user and the computer system that autonomously creates and executes plans for getting the bibliographic information while the author concentrates on writing. The collaborative model of human-computer interaction implemented in Writer's Aid places new requirements on the planner, which must posess the expressive power to formulate complex planning goals, efficiently represent and reason about an incompletely specified domain, interleave planning with execution of partial plans, and dynamically re-plan in case of execution failure while avoiding redundant information gathering, all in real time.

In the demonstration the participants will be able to use the system and evaluate its unique design in which information integration occurs within a context of a writing task that is the source of the user's information need.

1 User Interactions

Writer's Aid is activated when an author working on a LaTeX document in Emacs text editor inserts a citation command inside the body of the document. In the citation command the author specifies the keywords that identify the potentially relevant papers, and whether s/he is interested in obtaining the viewable version of the papers or the bibliographic records only. While the author continues writing and inserting citation commands, Writer's Aid dynamically creates and executes plans for obtaining the bibliographic records from the available author's personal bibliographic collections and preferred on-line collections [1] such as ResearchIndex or ACM Digital Library, caching viewable versions of papers and, upon the user's request, presenting the results of the search in a formatted list. At this point, the author may select a paper reference from the list, and the system will make sure that the citation is ready for compilation of the document's reference list.

While planners have been widely used in information integration systems (e.g., [Barish *et al.*, 2000; Lesser *et al.*, 2000]) the distinguishing feature of Writer's Aid is that it supports a simultaneous ongoing user's activity that is the source of the information request. A smooth integration of the search and selection of cited papers with the process of writing is a key requirement on Writer's Aid's interface: the user can insert new citation commands and access possibly incomplete results of the search for any of the citation commands at any time while editing the paper.

2 System Architecture

Figure 1 depicts the key components of the system. The Goals database records Writer's Aid's planning goals, while The State Of Knowledge (SOK) database records its knowledge about papers, their locations, user preferences, and other related domain information. Both databases use the PSIPLAN [Babaian & Schmolze, 2000] representation. The kernel of the system is a Planning Problem Manager (PPM) that uses the PSIPOP-SE algorithm to plan and execute the search for bibliographic information.

A user's request to obtain papers relevant to a set of keywords is formulated as three separate PSIPLAN goals that collectively can be summarized in the following statement

> *For each paper that is relevant to a set of specified keywords according to some local or remote bibliography preferred by the user, get that paper and get the bibliographic record for it.*

The Goal Reduction module picks up a goal posted to the Goals database and separates the part of the goal that is entailed by the SOK and thus is already accomplished, from the part that still remains to be achieved. The latter part is passed to the Planning Problem Manager, which creates an instance of a planning problem and hands it to the PSIPOP-SE planner. The planner constructs and executes a plan or

[1] The set of the preferred bibliographies is specified manually during the initial tune-up of the system to a new user.

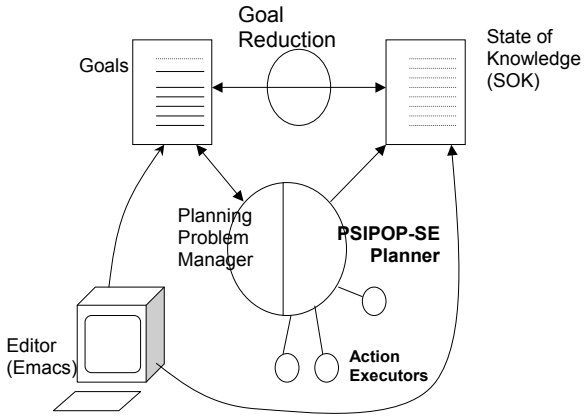

Figure 1: Components of the Writer's Aid's architecture.

reports failure if the planning problem is unsolvable. Upon executing the plan actions, the SOK database is appropriately updated to reflect all changes in the knowledge about the state of the world.

When a user issues a command to view a list of records and papers corresponding to a citation, this information is derived from the SOK, formatted, and presented in a separate window for browsing and selection.

3 Prioritizing and Reducing Goals

Writer's Aid starts out having no or little knowledge about the papers and their locations, and therefore substantial amount of information gathering from various on-line sources may be required to satisfy a user's requests. However, it is very likely that the author will be satisfied with only a subset of the search results. To guarantee a quick delivery of the most likely relevant results, the goals arising from the information requests are prioritized. For instance, papers and bibliographic records that are listed in the author's own personal collection are more likely to be cited and Writer's Aid will work on the goal of getting them first.

To make the partial results on *all* of the issued requests quickly available to the user, the planning goals are processed by the Planning Problem Manager in a round-robin fashion. As there may be significant overlap in the information related to different citation commands (e.g., location of the papers, authors' web sites) each goal posted in the goal database is reduced to account for already available information prior to being sent off to the planner. Such reduction separates the part of the goal that has not yet been achieved, ensuring more efficient operation of the planner. To further ensure non-redundancy of operation, all planning problems share the same database of knowledge, which is updated after each execution of an action to reflect the changes in the world and the Writer's Aid's state of knowledge.

4 Planning

Most of the Writer's Aid's planning problems do not have a solution plan that *provably* achieves the goal. It is possible, houwever, to construct the solution by interleaving the process of planning with execution of some information gathering steps. Combining planning with execution in PSIPOP-SE is implemented as a search in a space of *hypothetical plans*. Hypothetical plans are partial plans in which the achievement of the goal may be conditioned on certain outcomes of sensing. Hypothetical plans condition on the value of an unknown subgoal; by verifying a hypothesis via execution of a sensing action the planner eventually collects enough information and reduces the incompleteness of knowledge enough to find a solution plan or establish the goal to be unsatisfiable. For example, having no information on the location of a paper, the planner may adopt a hypothesis that the paper is available from a certain collection, and verify the information by querying that collection.

Given a rich set of Writer's Aid's sensing actions, the space of all hypothetical plans is very large and must be explored gradually to avoid combinatorial explosion resulting from a high degree of incompleteness. The parameter that PSIPOP-SE uses to control the size of the search space is called the *hypothetical level* of a plan; it is the maximum number of consecutive hypotheses supporting a single subgoal in a plan. PSIPOP-SE starts by considering plans with hypothetical level 0, that is, the classical plans with no hypothesis, and only increases the hypothetical level of considered plans after all plans with lower hypothetical levels have been considered and their hypothesis verified, a kind of *iterative deepening in hypotheticals*.

A more detailed description of the system and the results of a laboratory user study are presented in [Babaian, Grosz, & Shieber, 2002].

References

[Babaian & Schmolze, 2000] Babaian, T., and Schmolze, J. 2000. Psiplan: Open world planning with ψ-forms. In *Artificial Intelligence Planning and Scheduling: Proceedings of the Fifth International Conference (AIPS'00)*, 292–300.

[Babaian, Grosz, & Shieber, 2002] Babaian, T., Grosz, B. J., and Shieber, S. M. 2002. A writer's collaborative assistant. In *Proceedings of Intelligent User Interfaces'02*, 7–14. ACM Press.

[Barish *et al.*, 2000] Barish, G., Knoblock, C. A., Chen, Y.-S., Minton, S., Philpot, A., and Shahabi, C. 2000. The TheaterLoc virtual application. In *AAAI/IAAI*, 980–987.

[Lesser *et al.*, 2000] Lesser, V., Horling, B., Klassner, F., Raja, A., Wagner, T., and Zhang, S. 2000. Big: An agent for resource-bounded information gathering and decision making. *Artificial Intelligence* 118:197–244.