John S. Gero, Scott Chase and Mike Rosenman (eds), *CAADRIA2001*,
Key Centre of Design Computing and Cognition, University of Sydney, 2001, pp. 275-282.

AN APPROACH TO SEARCH AND EXPLORATION THROUGH MIXED-INITIATIVE

SAMBIT DATTA
Deakin University
Australia

AND

ROBERT F. WOODBURY
Adelaide University
Australia

Abstract. Generative design environments need support for human intervention as well as sound computational formalisms. A systematic approach to integrating the two, formal generation and the exploratory, is lacking. In this paper, we posit the possibility of a design support system that combines formal search with user driven exploration. Our approach is to cast the interaction between the user and the generative formalism as agent collaboration in a mixed-initiative environment. We describe the role of interaction and agency in an experimental mixed-initiative design support system, *FOLDS* and demonstrate its application.

1. Introduction

Computer-based environments for supporting design are complex software artifacts. These tools need to use sound computational formalisms as well as address issues of human usability. The development of interactive and usable *generative systems* is a significant research area in design computation (Carlson, 1993; Chein 1998; Harada 1997; Heisserman, 1991). They apply computational formalisms to guide, reason about and support the task of designing, usually in loose association with human intervention. Though classical search techniques play a central role in the generative kernels of these "closed-world" systems, the open-ended exploration of design spaces is the desirable goal. We view the role of the user as central to the task of exploration.

Search-based strategies, derived from problem-solving techniques have been deployed in the early research systems as they were modelled on expert systems. Search is characterised by the use of global control strategies and sequential models of execution. The literature on the theory of search and problem-solving has established a number of key concepts useful in design.

The notions of *state*, *move* and *space* are essential in describing compter-based environments. However, the characteristics of search that make it attractive to problem-solving have been found to be problematic in the activity of designing (Gero, 1994; Smithers et al 1994).

Designing is limited by global control policies, and is not a sequential predictive process. Designing cannot be reduced to search, and requires a broader and more inclusive notion of design support. Smithers et al (1994, 1998) develop *design as exploration*, a knowledge level theory of design process based on the understanding of ill-structured problems in general and design problems in particular. They characterise ill-structuredness as incomplete, inconsistent, imprecise and ambiguous. These properties require exploration, and are not amenable to search. They define exploration, as

> *"involving the construction and incremental extension of (well-structured) problem statements and associated solutions....devising intermediate problems, at "tangents'" to the main design problem'" (Smithers 1994, page 303—304).*

We contend that while this is true, search and exploration are not the oppositional paradigms they are made out to be. They do not represent polar positions, but function as complementary strategies in design support environments. Hence, to provide effective design support, it is necessary to combine their complementarities through some model of collaboration.

In this paper, we sketch a model of interaction, that casts the interplay between search and exploration as agent collaboration in a mixed-initiative environment. We conceptually distinguish between search and exploration as strategies employed by two agents, one explorative and the other search-based. We develop the proposition that *mixed-initiative* can combine search and exploration in computer-based environments for designing.

Mixed-initiative provides an effective interaction model based on the dynamic sharing of initiative and control during a collaborative session between two or more agents (Allen, 1994; Burstein et al 1999; Veloso et al 1997). This notion of initiative can be used to develop a model of interaction, in which a human user can collaborate with a software agent to solve a problem.

In order to investigate this proposition, we present an account of our computer-based design support environment, *FOLDS* (Datta, 2000). We show how a formal resolution technique (Burrow et al, 1999; Woodbury et al, 1999) is adapted for open ended user exploration to develop an understanding of movement in design spaces. We present an outline of the system and examine the affordances of interaction and agency in such an environment.

2. Supporting Design Space Exploration

The ideas sketched here in broad terms are based on the logic of typed feature structures (Carpenter 1992). The reader is referred to the formal research reported on design space exploration (Burrow, 1999; Woodbury 1999; Woodbury 2000; Datta 2000) for a detailed exposition on the topic. *Feature structures* are much like the frames of AI systems, the records of imperative programming languages like C or Pascal, and the feature descriptions used in standard linguistic theories of phonology and syntax. They provide the theoretical basis for developing the formal machinery necessary for describing design space exploration (Woodbury et al, 2000).

2.1 SUBSUMPTION AND PI-RESOLUTION

A typed generative system comprises the following :

- A type system, comprising an inheritance hierarchy of types, features and constraints expressed in a description language,
- a formally structured design space, comprising discrete states that contain typed feature structures, partially ordered by the *subsumption* relation and
- a set of formal operators for generating states based on the description language using a constraint resolution technique called *pi-resolution*.

Pi-resolution (Burrow, 1999; Carpenter 1992) is the fundamental feature structure construction process. It is a non-deterministic search for solutions to a query description. A solution is represented as a feature structure, which satisfies a description and the constraint system. The type system is composed of an inheritance hierarchy of types, features and constraints. It is used for modeling conceptual information about empirical objects (such as attributes and values) and for encoding design knowledge. The generator is a statement drawn from a description language. The design space is the formal space of partial satisfiers of the query, constructed with respect to the generator.

2.2. CRITERIA FOR INTERACTION

In our current consideration of mixed-initiative, the agent definition is limited to the human designer and the system. Admittedly, this is an unsatisfying level of agent granularity, given that systems have widely differing capabilities and humans even more so. However, the agent distinction is sufficient to enable us to frame the problem of search and exploration in machine-supported design interaction.

Allen (1994) identified three main aspects of a mixed-initiative approach involving two or more interacting agents, namely,

- flow of *control* : Can the agents acquire and relinquish control of the dialogue ?

- share and maintain *context* : Are the agents able to communicate successfully ?
- *focus* of attention : Do they consider the same task at hand ?

The Model-View-Controller (MVC) framework provides a natural paradigm for supporting the above criteria. It espouses a partitioning of the application domain specification (model), the graphical interface specification (view) and the interactivity specification (controller). The benefit of adopting the model-world approach is that search and open-ended exploration can coexist, the active components of the model can be treated as tools with little or no autonomy while the system as a whole may have considerable autonomy in ordering the environment to facilitate problem-solving.

In addition to the above, we specify three major criteria in developing a mixed-initiative design-support system, namely,

- to support the compilation and modelling of design knowledge in the application domain (the domain level),
- to support the incremental completion of design tasks (the task level),
- to support user interfaces for facilitating dialogue with the formalism (the dialogue level)

Next, we investigate how mixed-initiative enables the integration of search and exploration as discussed above at the task level in greater detail.

3. Mixed-Initiative in FOLDS : The Task Level

Mixed-initiative at the task level in *FOLDS* is characterised by :

- the incremental pi-resolution process as the search mechanism in the interaction loop,
- display of the intermediate states of computation to exploration by the user and
- the incorporation of limited direct *manipulation* capabilities in the display of feature-value graphs, satisfier nodes their graphical representation using the model-view-controller approach.

3.1. INCREMENTAL PI-RESOLUTION

Incremental Pi-resolution (Burrow 1999) provides a natural entry point for human-computer interaction in the resolution process. The incremental process allows a two-stage dialogue between the designer and the resolution process.

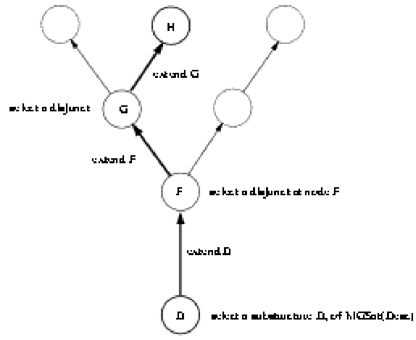

Figure 1. A schematic representation of resolution.

First, given a query description, the explorer traces a path through the sub space of possible states that are consistent with the description and presents the results for user intervention. The designer then resolves the non-determinism in the operation and presents the explorer with the next legal operation.

Second, the explorer retrieves the last element in the path and presents its root node to the user. The designer then chooses one of the sub nodes of the structure and requests an incremental pi-resolution operation on this structure. The explorer then builds the set of types and associated recursive type constraints to which the selected node can be refined and presents them to the user. If the structure is fully resolved, the explorer marks the structure as resolved with respect to the type constraints, and requests the user for the next operation. Given a set of legal types, the designer can choose one of the subsuming types and present the explorer with a legal operation.

In both scenarios described above, the incrementality of the generative process permits the designer to decide what the next forward operation in the satisfier space should be and thus have local control over the paths. Design decisions are not subject to some global inference strategy, but are goal-directed. And each resolution step introduces new constraints and opens up possible spaces via designer-explorer interaction.

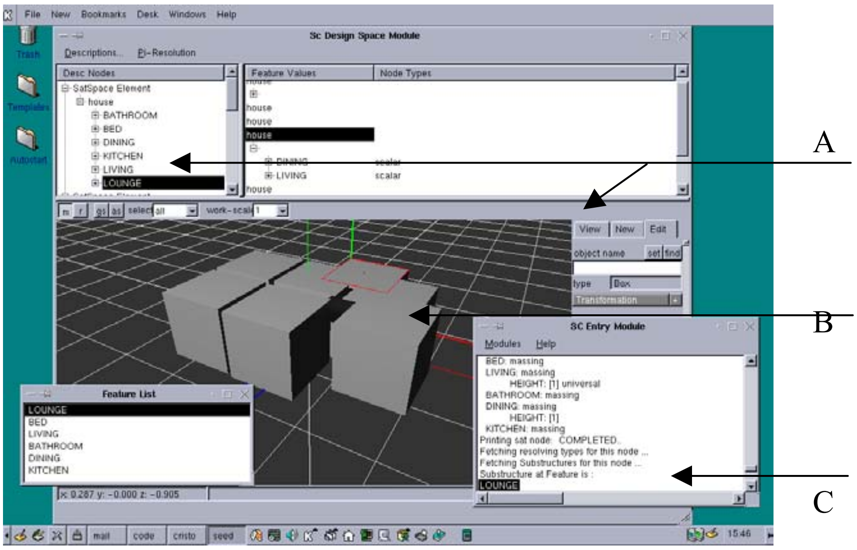

Figure 2. Mixed-initiative in *FOLDS*. The system returns attribute-value matrices (A) and presents a graphical view (B) and context information (C) relevant to an individual state.

3.2. DISCUSSION

The premises of the mixed-initiative dialogue in *FOLDS* are :

- A two-stage dialogue with the formal resolution process,
- uniform display of the intermediate states of computation, e.g.descriptions, satisfier and states as foldable attribute-value matrix (AVM) nodes and
- direct *manipulation* of the AVM and their graphical representations, using a model-view-controller approach.

We discuss each of these aspects with reference to control, communication and focus of attention issues in agent interaction. First, the *incrementality* of the search process enables the user to step into the computation during resolution and take control of the non-determinism. Second, the user can also communicate the next step of resolution or shift focus to other parts of the space through the display of the *intermediate* states of computation. Finally, the display of these states as folded feature-value graphs provides a compelling metaphor for *manipulation* and control of the explored space.

Consequently, a user interacts with the formal generative operators of the explorer and constructs a *design space* recording the states. In our view, while the operators have a formal basis, the exploration process itself is open-ended. Hence, the user can control the generation of partial designs and maintain the formal integrity of the resolution process.

5. Conclusion

The use of mixed-initiative interaction presents an effective approach for combining a search-based agent and open-ended exploration in the domain of design space exploration. These paradigms are viewed, not as mutually exclusive, but interrelated and integral to design support systems. This paper explores the possibility of combining formal search mechanisms and open-ended exploration models in design. It presents a mixed-initiative model in *FOLDS* wherein human designers can collaborate with software agents in an interactive environment. This environment illustrates the interleaving of mixed-initiative concepts such as *turn-taking*, *control, communication and focus of attention.* testing the relationship of search and exploration

Acknowledgements

The authors acknowledge the anonymous reviewers for their comments and suggestions on this paper. Andrew Burrow, Teng-wen Chang and other members of the SEED research group in Adelaide and Pittsburgh who participated in discussions on the topic and contributed to the development of the ideas.

References

Allen, J. F.: 1994, Mixed-initiative planning, *ARPA/Rome Labs Planning Initiative Workshop.* Position Paper.

Burrow, A. and Woodbury, R. F.: 1999, Pi-resolution and design space exploration, *Proceedings of the Eighth International Conference on Computer-Aided Design Futures*, pp. 291--308.

Burstein, M., Mulvehill, A. and Deutsch, S.: 1999, An approach to mixed-initiative management of heterogeneous software agent teams, *Thirty-second Annual Hawaii International Conference on System Sciences,* Maui, Hawaii.

Carlson, C.: 1993, *Grammatical Programming: An Algebraic Approach to the Description of Design Spaces,* PhD Thesis, Department of Architecture, Pittsburgh, Carnegie-Mellon University.

Carpenter, B.: 1992, *The Logic of Typed Feature Structures with Applications to Unification Grammars, Logic Programs and Constraint Resolution*, Cambridge University Press.

Chien, S.-F.: 1998, *Supporting Information Navigation in Generative Design Systems*, PhD Thesis, School of Architecture, Carnegie-Mellon University.

Datta, S. and Woodbury, R.: 2000, Proactivity and mixed-initiative in unfolding design spaces, *in* K. Bennett, and S. Goss, (eds), *Workshop on Situated Activity Of Entities (Agents/Humans) Within Complex, Dynamic, Real-Time Environments (Real/Simulated)}*, *OZCHI2000 : Interfacing Reality in the New Millenium*, pp. 15-16.

Gero, J. S.: 1994, Towards a model of exploration in computer-aided design, *in* J. S. Gero, and E. Tyugu (eds), *Formal Design methods for CAD, TC5/WG5.2 Workshop on Formal Design methods for CAD*, pp. 315--336.

Harada, M.: 1997, *Discrete/Continuous Design Exploration by Direct Manipulation*, PhD Thesis, Department of Architecture. Pittsburgh, Carnegie Mellon University.

Heisserman, J.: 1991, *Generative Geometric Design and Boundary Solid Grammars*, PhD Thesis, Department of Architecture. Pittsburgh, Carnegie Mellon University.

Smithers, T., Corne, D. and Ross, P.: 1994, On computing exploration and solving design problems, *in* J. S. Gero, and E. Tyugu (eds), *Formal Design Methods for CAD, TC5/WG5.2 Workshop on Formal Design Methods for CAD*, pp. 293--313.

Smithers, T.: 1998, Towards a knowledge level theory of design process, *in* J. S. Gero and F. Sudweeks (eds), *Artificial Intelligence in Design '98*, Kluwer Academic Publishers, Dordrecht, pp. 2-22.

Veloso, M., Mulvehill A. and M. Cox.: 1997, Rationale-supported mixed-initiative case-based planning, *Innovative Applications of Artificial Intelligence,* pp. 1072--1077.

Woodbury, R., Burrow, A. Datta, S. and Chang, T.: 1999, Typed feature structures and design space exploration, *AI in Design, Engineering and Manufacturing* **13**(4), 287-302.

Woodbury, R., Datta, S. and Burrow, A.: 2000, Erasure in design space exploration, *in* J. S. Gero (ed.), *Artificial Intelligence in Design '00*, Kluwer Academic Publishers, Dordrecht, pp. 521-543.

.