

## Animation of Horn Clause Proofs

Kenneth M. Kahn  
Xerox PARC  
3333 Coyote Hill Road  
Palo Alto, CA 94304  
kahn@parc.xerox.com

Janus, a concurrent constraint programming language, provides an efficient and expressive means of constructing distributed and parallel computations [5]. Concurrent logic programming languages [4] are a special case of concurrent constraint programming languages in which the constraint system is confined to the Herbrand domain. Concurrent logic programs are collections of Horn clauses with control annotations. The execution of a terminating concurrent logic program can be viewed as a proof of a query using the program as a set of axioms. The implementations of such languages can be seen as sound but incomplete theorem provers.

Pictorial Janus was built to support a pictorial syntax for the Janus programming language [2, 3]. [1] explores the ways in which Pictorial Janus, while constructed to visualize concurrent constraint programs and their executions, can be used to automatically produce animated illustrations of proofs. Furthermore, theorems can be stated in terms of diagrams which are automatically proved and animated. The diagrams are capable of expressing Horn clauses with variables and terms. The meaning of such a diagram relies only upon the topological properties of the drawing. Shapes and colors are used to improve readability and aesthetics, but play no semantic role.

A component of the Pictorial Janus system is a parser of PostScript descriptions of drawings. Thus, theorem diagrams can be constructed using any illustration tool capable of producing PostScript output. Using only this output, the parser discovers what elements are variables, terms, clauses, and so on. This information, along with the original choice of shapes, colors, layout, and so on, is passed to the animation component, which attempts to prove the query. If successful, the result is an animation of the proof which maintains the graphical style and conventions of the original diagrams. Such animations, since they exploit the abilities of the human visual system, are potentially far easier to understand than conventional proofs.

A variant of the current system containing a complete Horn clause theorem prover would be even more useful, and experience with the construction of Pictorial Janus suggests that such a variant would be easy to construct.

## References

- [1] Ken Kahn. Animation of horn clause proofs. Technical report, Xerox PARC, 1992.
- [2] Ken Kahn. Concurrent constraint programs to parse and animate pictures of concurrent constraint programs. Technical report, Xerox PARC, 1992. Will appear in FGCS92.
- [3] Kenneth M. Kahn and Vijay A. Saraswat. Complete visualizations of concurrent programs and their executions. In *Proceedings of the IEEE Visual Language Workshop*, October 1990.
- [4] Vijay A. Saraswat. *Concurrent constraint programming languages*. Doctoral Dissertation Award and Logic Programming Series. MIT Press, 1992.
- [5] Vijay A. Saraswat, Kenneth Kahn, and Jacob Levy. Janus—A step towards distributed constraint programming. In *Proceedings of the North American Logic Programming Conference*. MIT Press, October 1990.