A Library of Agents for Learning Environments

P. Dillenbourg, P. Mendelsohn, M. Hilario, D. Schneider, B. Borcic
TECFA, Faculté de Psychologie et des Sciences de l'Education
Université de Genève, 9 Route de Drize, 1227 Carouge, Switzerland
E-Mail: pdillen@divsun.unige.ch

Most authoring shells include pedagogical knowledge for planning instruction at a macro-level (selecting curriculum units). Little attention has been paid to the knowledge necessary to monitor and to regulate problem–solving activities at a micro-level. We have implemented two systems in which we focussed on this issue. MEMOLAB is an interactive learning environment (ILE) in experimental psychology (Dillenbourg & Mendelsohn, 1992). ETOILE is a toolbox for creating ILEs such as MEMOLAB. It includes several domain–independent agents (tutors) that will interact with the domain–specific agents (experts) developped by the designer.

Domain Independent Teaching Styles

Monitoring problem solving activities requires a large amount of domain–specific knowledge and may appear to contradict the quest for domain–independent pedagogical expertise. We can bypass this contradiction if the tutor does not directly interact with the learner, but instead monitors the interaction between the learner and an expert. This expert is domain–dependent, but its interactions with the learner can be described at a higher level, with a domain-independent language: e.g. "the learner asks the expert to continue", "the expert does not understand what the learner did", "the learner does not want the expert to change what he did", ... The tutor observes the expert-learner interactions in order to take a decision. For instance, when none of its activatable rules matches what the learner did, the expert tells the tutor he disagrees. Then, the tutor may ask the expert to suggest another action, he may open an hypertext node (Memolab includes two hypertexts) or he may simply let the learner continue. The tutor's decision depends on his 'teaching style' (Elsom-Cook, 1991), a style implemented as a set of rules. ETOILE includes several tutors (Skinner, Bloom, Vygotsky, Piaget and Papert). For instance, the tutor 'Vygotsky' initially asks the expert to do most of the work and then 'Vygotsky' progressively increases learner participation. 'Bloom' allows the expert to correct the learner while 'Papert' does not.

Implementing Expert-Learner Collaboration

Teaching styles constitute variations of a default paradigm in which the learner and the expert solve problems jointly and incrementally. Our previous research on human-machine collaboration (Dillenbourg & Self, 1992) taught us that collaboration works better when it concerns concrete objects rather than abstract knowledge. In ETOILE, the learner and the expert share a set of commands for transforming the concrete representation of the problem state. The expert rule conditions directly refer to the displayed objects. Thereby, the expert is aware of any transformation performed by the learner. Conversely, expert rules modify the problem state display and thereby inform the learner of the expert actions. This supports a fine-grained and opportunistic collaboration. The expert and the tutor rulebases are implemented as object-oriented production systems (OOPS). The set of facts is the set of instances of the classes referred in the system rules. The engine and ETOILE are implemented in object-oriented Common Lisp (CLOS), the interface part is written with the Common Lisp Interface Manager (CLIM). ETOILE runs on Sun Sparcstations.

References

Dillenbourg, P. & Mendelsohn, P. (1992) The genetic structure of the interaction space. In E. Costa (Ed.)*New Directions for Intelligent Tutoring Systems* (pp.15–27) Berlin: Springer–Verlag

Dillenbourg, P. & Self, J.A.(1992) A computational approach to socially distributed cognition.*European Journal of Psychology of Education*, 7(4), pp. 353–372.

Elsom–Cook, M.T. (1991) Dialogue and teaching styles. In P. Goodyear (Ed.)*Teaching knowledge and intelligent tutoring* (pp.61–84) Norwood, NJ: Ablex.

Acknowledgements

The work is funded by the Swiss NSF research program on AI and Robotics, grant 4023–27013.