Producing Non-verbal Output for an Embodied Agent in an Intelligent Tutoring System

Roger Nkambou[1] and Yan Laporte[2]

[1]Département d'Informatique
Université du Québec à Montréal, Montréal, H3C 3P8, Canada
Nkambou.roger@uqam.ca
[2]Département de Mathématiques et Informatique
Université de Sherbrooke, Sherbrooke, J1K 2R1, Canada
ylaporte@acm.org

Abstract. This paper discusses how we can generate non-verbal output through an embodied agent from a user's actions in an ITS. The most part of the present work is related to maintaining an emotional state for a virtual character. We present the basic emotional model we used for internal emotions management for the character. Then we follow with an overview of the agent's environment followed by the role he is designed to play using our own system as a reference. We then give an overview of its internal architecture before we move on to what are the inputs taken by the system and how those are treated to modify the emotional model of the agent.

1 Introduction

Intensive researches have been done over the years on the development of believable agents, embodied agents or real time animated characters [1, 2]. These works have so far achieved very convincing results and impressive demonstrations. Some of them has been done in the field of ITS while the others are more in the interest of cognition theory and adaptive interfaces. Both of those fields are tightly linked to the development of ITS. The current work aimed at making agents being somewhat empathic by giving them the capability to guess the mood and emotional state of the user. For this purpose different models have been created to represent the emotions of the user.

This paper discusses how we can generate non-verbal output though an embodied agent from a user's actions in an ITS. The most part of the present work is related to maintaining an emotional state for a virtual character. We present the basic emotional model we used for internal emotions management for the character. Then an overview of the agent's environment is presented followed by the role the agent is designed to play using our own system as a reference. We then give detail of its internal architecture before move on to what are the inputs taken by the system and how those inputs are computed in order to modify the emotional model of the agent.

V.N. Alexandrov et al. (Eds.): ICCS 2001, LNCS 2074, pp. 366–376, 2001.

2 A Model to Represent Emotions

Among the various cognitive models proposed to represent emotions, one that has been used many times (including in the Oz project [3]) is the Ortony, Clore & Collins (OCC) model [4]. This model has been used by Elliot [5] to become the model presented in table 1. This model was usually used to depict the emotional state of a

Table 1. The OCC Model as described by Clark Elliot 1997

Group	Specification	Category Label and Emotion type
Well-Being	Appraisal of a situation as an event	**Joy**: pleased about an event **Distress**: displeased about an event
Fortunes-of-Others	Presumed value of a situation as an event affecting another	**Happy-for**: pleased about an event desirable for another **Gloating**: pleased about an event undesirable for another **Resentment**: displeased about an event desirable for another **Jealousy**: resentment over a desired mutually exclusive goal **Envy**: resentment over a desired non-exclusive goal **Sorry-for**: displeased about an event undesirable for another
Prospect-based	Appraisal of a situation as a prospective event	**Hope**: pleased about a prospective desirable event **Fear**: displeased about a prospective undesirable event
Confirmation	Appraisal of a situation as confirming or disconfirming an expectation	**Satisfaction**: pleased about a confirmed desirable event **Relief**: pleased about a disconfirmed undesirable event **Fears-confirmed**: displeased about a confirmed undesirable event **Disappointment**: displeased about a disconfirmed desirable event
Attribution	Appraisal of a situation as an accountable act of some agent	**Pride**: approving of one's own act **Admiration**: approving of another's act **Shame**: disapproving of one's own act **Reproach**: disapproving or another's act
Attraction	Appraisal of a situation as containing an attractive or unattractive object	**Liking**: finding an object appealing **Disliking**: finding an object unappealing
Well-being /Attribution	Compound emotions	**Gratitude**: admiration + joy **Anger**: reproach + distress **Gratification**: pride + joy **Remorse**: shame + distress
Attraction/ Attribution	Compound emotion extensions	**Love**: admiration + liking **Hate**: reproach + disliking

user and sometime try to infer what his state will become after a given episode. From this model, we can easily define couples of contradictory emotions [6] as indicated in table 2.

What we intended to do in our system is to use this model as an internal model for our agent and not for the user [7]. This practice has been discussed and judged flawed by the original authors [3] mainly because of the limited perception and judgment of a computer system compared to a human being. While this note was stated about inverting a model established from human behavior, which is unlike in our system where the model is driven by arbitrary rules, it is partly addressed in our approach by using the model in a defined environment and with a specific role to play. While this is obviously imperfect, it provides us with a unified representation of the emotion status of our agent. Some work has been done in this regard by morphing different faces together.

Each couple of emotions is represented by a single numerical value that can vary around a central point 0 being the absence of any imbalance in that emotion couple. Increasing the value of a given emotion will, at the same time, decrease the value associated to its counterpart.

Table 2. The Emotion Couples derived from the OCC model

Joy/Distress	Happy-for/Resentment	Sorry-for/Gloating	Hope/Fear
Satisfaction/ Disappointment	Relief/Fears-Confirmed	Admiration/Reproach	Pride/Shame
Liking/Disliking	Gratitude/Anger	Gratification/Remorse	Love/Hate
Jealousy/-Jealousy			

3 Defining the Environment

The Intelligent Tutoring System used as a test environment is comprised of a collection of virtual laboratories (some of them in 3D) where the student must perform tasks and solve problems using interactive simulations [8]. The learners are undergraduate university students and all the activities are part of different scientific curriculum. The emphasis of this system is on simulations lab equipment to accomplish tasks that would otherwise need to be done in a laboratory. This is done in order to allow students to virtually manipulate equipment by themselves and this practice provides a way to overcome problems related to the availability and high cost of the equipment.

The system activities range from 3D simulations of the equipment to more standard interfaces to accomplish tasks such as solving Logic problems in a way where it can be followed by the system. The user also uses the system to view the documentation associated with the activities. The realm of observable events for the agent is contained within those boundaries of consulting documentation and manipulating the virtual laboratories to accomplish given activities and tasks. A structure annotated with domain knowledge or at least a detailed description of what is available to the agent is associated with each activity.

All the information contained in the user model is available as well. The user model [9] is comprised in three parts: the cognitive model, the behavioral or affective model and an inference engine. The entire curriculum is mapped with a structure of underlying concepts which make it possible to specify the required concepts of a given activity and the ones acquired through a given step of the activity. Even particular steps or actions in an activity can be mapped. In the student model, each concept is associated with a mastering level and the source of this belief (the system, the inference engine, the user himself etc.).

The agent doesn't have any perception of anything outside this closed realm. While this limits its capacity to provide appropriate feedback, the user expects it to be that way because the agent is the agent for the tutoring system and nothing else.

4 Defining the Role of the Agent

Eventually, the agent will play the role of a coach in the system providing comments, critics, help and insights. Right now since the system is still in its first version, the goal set for the agent is to provide continual non-verbal feedback on the user's actions (or inactions). It acts more as a companion providing a dynamic element not to replace but to temporarily palliate to the lack of human participation which is a strong element of laboratory activities in the real world. This companion is a companion in the sense of another student [10] but rather a silent coach or maybe a strong student [11] nodding or twitching to the learner's doing. This allows mostly unobtrusive feedback to occur. The agent is also reflects the internal state of the system trying to track what the user is trying to do.

There are a number of concerns that are usually part of designing an embodied agent that must be addressed before defining the system. Most of these are given fixed value in the context of an ITS. The motivation of the agent will not change in our system, it remains at all times that the users acquires new knowledge and completes activities successfully. The "another" mentioned in the definition of the different emotions will always correspond to the learner. Also the attractive and unattractive objects will always be associated with learner's success and learner's failure in doing an action. The different event will however include actions that cannot lead directly to failure or success such as consulting a piece of media or the intermediate steps between two points where we know if the user is successful or unsuccessful. Events can also be constituted of user's inaction.

Also, the agent acts in his own reserved space and doesn't have to do any direct manipulation in the world at this point[2].

5 Architecture Overview

The agent (called Emilie) is designed in three layers each abstracting a part of the problem of generating movement in the agent to provide a response to events in the system. These events are exchanged in the form of messages. The interface is written in Java using the Java3D library while the other layers are implemented in an expert system. An expert system seemed appropriate because it has been used to model emotions in the past [12] and is relatively easy to program.

The interface layer is concerned with managing the 3D geometry and generating the movements from the actions issued from the motor layer. These commands are simple but still carry a lot of geometric information related to the virtual world. Examples of such commands are *lookAt(), setSmileLeft(), setSmileRight, setMouthOpeningCenter(), setLeftEyelid()* and so on. While most of them operate with relative parameters which are independent from the geometrical world, some, such as *lookAt()* must provide much more dependent information, in this case, a point in 3D space to look at.

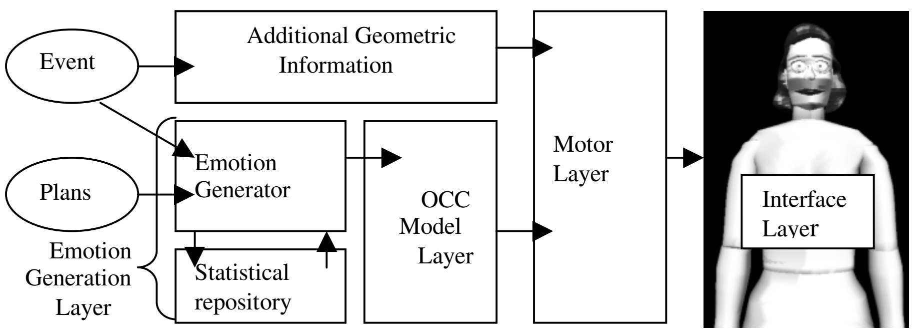

Fig. 1. Overview of the system

5.1 The OCC Model

The OCC model is the emotional model presented earlier in this article. It is implemented as a set of slots in the expert system with rules making sure it has both a ceiling and a floor so the values can't go off the scale. This is done to prevent a user to make the agent so angry or so happy that any changes afterward would be nearly imperceptible. The last set of rules is fired on periodical time intervals to bring the values closer to zero every time. These mechanisms are in place because no matter by which event the emotion was caused, the agent should eventually "get over it" and resume his earlier attitude with time.

5.2 The Statistical Repository

It contains useful information to generate an appropriate response. It stores information such as how many times was the user out of a known to be right path in a row. How many times was the user out of the known to be right path in a row the last time? Is the user on a streak correcting his past mistakes? How wrong the user usually is? What is the ratio of the user being right so far? Has the system considered a past stream of events from the student false while it led to the right solution earlier in this activity? These are important information to achieve coherent emotional response. If the user often uses path to the solution that are unknown of the system all the time and the agent looks angry every time, it is not a desirable behavior. An acceptable behavior would be to look confused or interested in the user's doing until we can evaluate what he is doing. Also if the user has a long streak of events where he is correct, we can't suddenly be angry or have high disapproval on a single mistake.

5.3 The Emotion Generator

The emotion generator is the part of the system that takes events and the planned actions in the system and generates modifications to emotion model. It takes into account information in the statistical repository to adjust his variations before sending them to the OCC model. The modifications made to the OCC model are in the form of decreases or increases of the values for the different emotion couples.

5.4 Additional Geometric Information

This part of the agent content information such as position on the screen of the latest actions. This information can be directly sent to the motor layer and is not related to the other layers anyway since their contribution is essentially in the real of the geometrical representation of the situation.

6 Choosing and Treating the Input

Since the information treated inside an ITS is very varied and comes in very large quantity, we had to define what was relevant to know to be able to generate changes to the emotional model. A large part of this effort is to reduce the amount of information to be taken into account to come up with a character showing satisfying behavior. First, all the information for each event is not relevant to the embodied agent. Domain tied knowledge is nearly unusable for his purpose. The information we retained for all the events coming from the user's are eventually brought down to two parameters: the degree of difficulty of the action accomplished (or to be accomplished in a case of failure) and the degree of "wrongness " of the student or rather how far his action is compared to the expected one (the right action having a value of zero). Another parameter carried in an event but unused by the emotion generator layer is the geometrical coordinates of where the event occurred, this is sent directly to the motor layer as an additional geometric information. Another type of event but which is treated differently is when the user undoes his last actions. This last category of action doesn't carry any parameters.

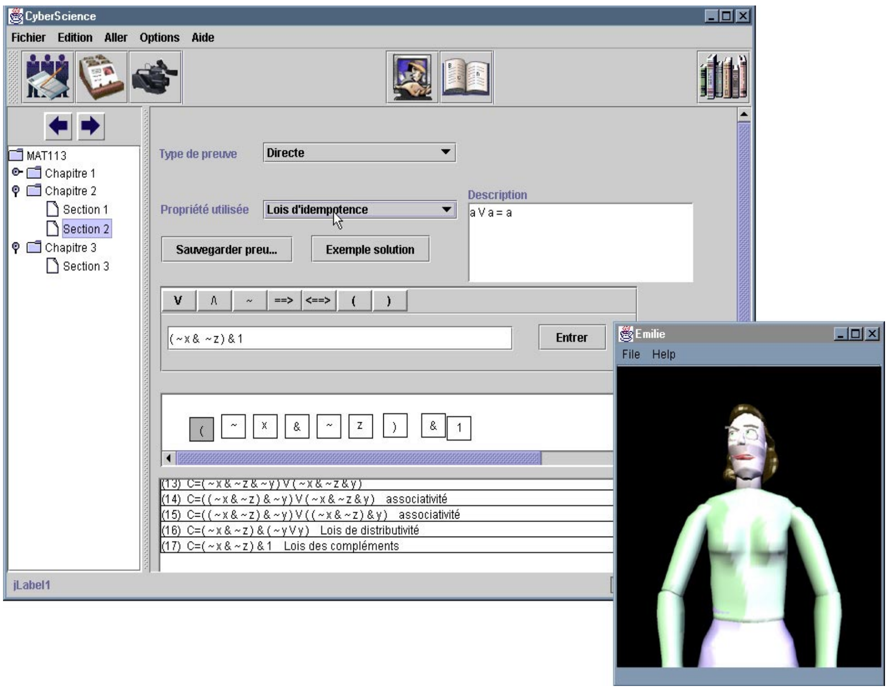

Fig. 2. The user just did something which is not part of the solution

The plans received by the emotion generation layer are merely a list of the difficulty factors that should be encountered by the student before he completes the activity. This information is to be used to produce hope, fear Satisfaction, Relief, Fears-confirmed, Disappointment.

The system also uses a lot of information that comes from the user model. This information is used to create a realistic degree of difficulty for a step or an event. The user model contains the information about the concepts mastered by the learner. In the curriculum we have prerequisites and concepts associated to the different steps in an activity and for the activity itself.

To determine the difficulty level associated with an activity or a step within this activity, we first retrieve an initial difficulty factor set by the designer of the activity. Since arbitrary values or values obtained by statistic about how difficult is an activity are unlikely to reflect the difficulty encountered by any given learner. To come to a better evaluation, we make adjustments based on information available in the curriculum and in the user model. If the student masters all the associated concepts well over the degree required to perform the step or the activity, then we set the associated difficulty level to a lower value. If the episode (the context in which the degree of mastery of a student on a concept was determined by the system) is judged similar to the current one, the difficulty level is set even lower because we consider that the user has already performed a similar task. If however the user has borderline or even incomplete requirements to be able to succeed in the activity, the difficulty factor is raised to a higher value. Also if the activity leads to a large number of new concepts, the degree of difficulty is set higher because we assume that the activity involves a lot of thinking and reasoning for the student to discover these new concepts.

The retrieval of a value to the parameter indicating how far is the student from the expected action is of course very domain related. There are however a few guidelines that can be established. First if the user is doing the expected action, the value should be of zero. If the user is doing something unexpected but that is not identified as a mistake, the value should be small. If the user makes a known mistake then the value should be higher.

7 Converting Events in Emotional Changes

After we have extracted the information about how difficult is the step associated with the incoming event and how wrong was the student was we use a set of predetermined rules to fire procedures that will adjust the OCC model. As an example, an event said to be “good” will generate joy and happy-for emotions, these emotions will be stronger if the degree of difficulty associated was higher. But the rules can be richer and lead to a much more sophisticated emotional feedback. If the user was in a streak of “bad” events, we should introduce satisfaction or relief. If the user just accomplished successfully a step that was judged as the most difficult in the activity we should raise satisfaction and relief only if the user encountered difficulties. If this was the last step in an activity, similar variations should be made to the model. Relief could be introduced if the user took a long time before taking action. Admiration value can be raised if the user succeeded on the first try to a difficult step of activity. It is said the user will prefer be flattered as to be reprimanded [13] so responses to “bad” events should be generally weaker.

When modifications are made to the OCC model, the motor layer converts them in movements or expression for the agent. This conversion includes very simple rules like setting the smile in relation with the joy value. But can also allow more sophisticated actions such as playing little scenes or series of actions when an emotion factor had a strong variation in a short period or modifying the amount of time the agent is directly looking at the user or to the action on the screen. In Figure 2 for example, the user just performed an action that we know is a mistake by selecting an item in the drop down list while it should have chosen another one, to show this the user looks at the source of the event while having an expression cause by the sudden rise of disappointment and distress.

Another example of a result is visible on figure 3 where the sudden rise of relief, joy and admiration led to the visible expression. The facial expression is in majority due to the joy and happy-for values while the hand movement is a result of the important variation in the admiration and/or relief levels.

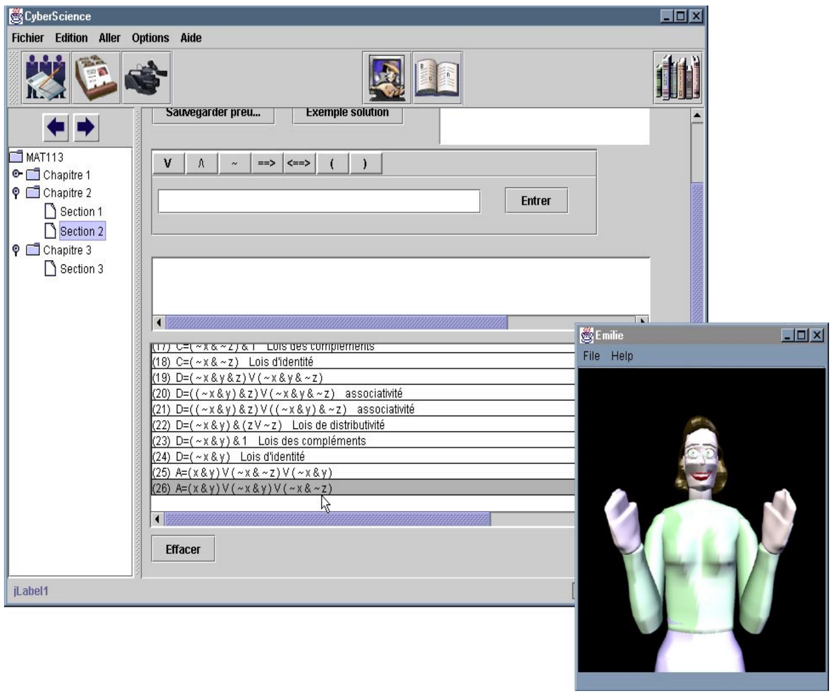

Fig. 3. The user just finished a difficult activity

8 Future Improvements

The current system uses the OCC model, which makes some expressions difficult to represent. For example it is still unclear how the system should represent confusion when the user accomplishes an unexpected action that is of an unknown value. Maybe there are more suitable models already existing or maybe we will have to tailor our own.

Numerous improvements could be made to the visual interface of the agent. Among them is the creation of a new 3D model which would be more cartoon like and require less polygons to draw, therefore offer better performances. Also another beneficial improvement would be that the agent would be in the same window where the activities take place. Adding transparency so the agent would look as it is drawn over the workspace would also make the system more usable. Better animation would also help to offer a better experience for the user.

Of course such a system would always beneficiate of a larger set of rules. Providing an authoring tool would most likely make the task of enlarging the current base easier. It would also offer better chances for our system to be reused in other contexts.

Some functionality will have to be integrated as well for the agent to really act as a coach. The agent should be able to converse with the user in some way. A text-to-speech feature would also improve the effect of the agent's feedback. For these features to be implemented, the agent will need to make a more important use of the other parts of the system such as the curriculum [14] or the representation of contexts.

It should also be able to choose appropriate materials from the media available when the user asks for help.

9 Conclusion

We proposed a way to generate simulated emotional responses to an embodied agent. Those visible responses are triggered by changes in an internal emotional model based on cognition theory. We have given examples of how information available in an ITS can be used to modify this model in response to events from the user. Such a system could be implemented in conjunction with other sources of actions to raise the quality of the interaction with embodied agents in Intelligent Tutoring Systems. We will continue to enhance this model and the implementation we are using. We are considering to eventually release the project and it's source code in hope that other people will use and enhance it.

10 References

1. Rickel, J. and Johnson, L. 2000. Task-Oriented Collaboration eith Embodied Agents in Virtual Worlds. In J. Cassel, J. Sullivan and S. Prevost (Eds.), *Embodied Conversational Agents*, MIT Press 2000.
2. Johnson, L., Rickel, J. and Lester, J.C. (2000). "Anumated Pedagogical Agents: Face-to-face interaction in interactive learning environments. *International Journal of Artificial Intelligence in Education.* 11: 47-78.
3. Ortony A., Clore G., and Collins A. *The Cognitive Structure of Emotions*, Cambridge: Cambridge University Press, 1988.
4. Bates J. The Role of Emotion in Believable Agents. Communications of the ACM, Speical Issue on Agents, July 1994
5. Elliott, C. 1992. *The Affective Reasoner: A Process Model of Emotions in a Multi-agent System*. Ph.D. Dissertation, Northwestern University. The Institute for the Learning Sciences, Technical Report No. 32.
6. Sassine Abou-Jaoude and Claude Frasson. *Emotion computing in competitve learning environments*. In Working Notes of the ITS '98 Workshop on Pedagogical Agents, pages 33--39, San Antonio, Texas, 1998.
7. Elliott, C.; Rickel, J.; and Lester, J. 1997. Integrating affective computing into animated tutoring agents. In *Proceedings of the IJCAI97 workshop, Animated Interface Agents: Making them Intelligent*, 113-121.
8. Nkambou, R., Laporte, Y. 2000. Integrating Learning Agents in Virtual Laboratory. In *Proceeding of World Conference on Educational Multimedia, Hypermedia & Telecommunication*, pp. 1669-1671. AACE.
9. Nkambou, R. 1999. Managing inference process in student modelling for intelligent tutoring systems. In Proceeding of the *IEEE International Conference on Tools with Artificial Intelligence, pp. 16-21.* IEEE press.
10. Aimeur, E., Dufort, H., Leibu, D., and Frasson, C. (1997). Some justifications for the learning by disturbing paradigm. In DuBoulay, B., Mizoguchi, R. (eds.), *Artificial Intelligence in Education: Knowledge and Media in Learning Systems. The Proceedings of AI-ED 97, Kobe, Japan, August 1997, 119-126.*

11. Hietala, P., Niemirepo, T. (1998). "Current Trends and Applications of Artificial Intelligence in Education. Presented at *The Fourth World Congress on Expert Systems*, Mexico City, Mexico.
12. Scherer, K. 1993. Studying the emotion-antecedent appraisal process: An expert system approach. *Cognition & Emotion* 7(3):325-356.
13. Fogg, B. J., Nass, C. 1997. Silicon sycophants: The effects of computers that flatter. *International journal of human-computer studies* 46(5):551-561.
14. Nkambou, R., Frasson, C. and Gauthier, G. 1998. A new approach to ITS-curriculum and course authoring: the authoring environment. *Computers & Educatio*n. 31: 105-130.