When H. C. Andersen is not talking back

Niels Ole Bernsen

Natural Interactive Systems Laboratory, University of Southern Denmark,
Campusvej 55, Odense, Denmark
nob@nis.sdu.dk, http://www.nis.sdu.dk

Abstract. The paper describes ongoing work on a spoken computer game system in which children, young people, and others can have domain-oriented spoken and gesture conversation with fairy-tale author Hans Christian Andersen (HCA) and some of his fairy tale characters. Having presented the scenario concept, the paper focuses on HCA character module management, discussing, in particular, how to manage HCA when he is not talking back to the user but is either alone in his study or receiving the user's conversational contribution.

1 Introduction

Beyond the current, *task-oriented* application paradigm for spoken language dialogue systems (SLDSs) [1] lie the *domain-oriented* SLDSs. Such systems have abandoned the task constraints which made SLDSs possible, but are still constrained by their domains. By contrast, humans are able to conduct unconstrained spoken conversation. For intelligent virtual agents to believably conduct domain-oriented conversation, they must behave naturally even when they do not address their interlocutors. This paper proposes solutions to the latter problem in the context of addressing the domain-oriented conversation challenge in the project NICE (Natural Interactive Communication for Edutainment, www.niceproject.org). Sect. 2 presents the NICE scenario and HCA. Sect. 3 describes the HCA character module. Sect. 4 presents our approach to the management of HCA as and when he is not generating a conversational turn.

2 The NICE Scenario, HCA's Personality and Knowledge

In the NICE scenario, 3D life-like fairy tale author HCA is in his study surrounded by artefacts, a writing desk and chair, an easy chair, and open floor space. At the back of the study, a door leads into the fairy tale games world populated by some of his fairy tale characters, such as the Naked Emperor and Cloddy-Hans. Talking to HCA, the user becomes an avatar who walks into HCA's study. The user can have spoken conversation with HCA and use 2D gesture input for indicating artefacts during conversation. At some point, the user is invited to go into the fairy tale world to engage in spoken computer games with the characters. The scenario of use in NICE is for the

system to be installed in public spaces, such as the HCA museum in his native city, Odense. The users will be visitors to the museum and are likely to spend limited time with the system, such as 5-20 minutes.

HCA's conversational ability derives from the domains he is familiar with, i.e.: (1) his eventful life and experience; (2) his fairy tales; (3) his person, perceived physical presence, study, and the artefacts present; (4) his role as "gate keeper" for access to the fairy tale games world; and (5) the user. HCA asks questions about the user and uses the information gathered during conversation. Based on the HCA literature and Wizard of Oz [1] data from spoken interactions between kids and a simulated HCA, each of HCA's conversational domains is decomposed into topics. In the user domain, for instance, the current topics are name, age, gender, go_to_school, origins, friends, games/plays_liked, travels, and favourite fairy tales.

Fig. 1. Hans Christian Andersen in his 19 Century study

HCA is in his 50s, quick-witted and imaginative, has a strong personal presence, often takes the initiative in conversation, is emotionally sensitive, and can be quirky at times, as when angrily refusing to discuss certain topics. HCA's personality is an important factor in shaping the users' conversational contributions. Whenever HCA takes the initiative, we may expect the user to respond in kind. This facilitates anticipation of the next user input and hence makes its processing easier. HCA's personality is reflected in the operations of the HCA character module (CM).

3 The HCA Character Module

The NICE system is Windows-based and has a client-server architecture based on partner Liquid Media's computer game engine FEngine. The system is being implemented in C++ and Prolog, using XML for data transfer. It uses late fusion for combining the speech and gesture input streams. The input fusion module sends a semantic representation to the relevant character module (CM). Following CM processing, a semantic output representation is sent to the response generation (RG) module which generates text-to-speech for the speech synthesiser and output to the display.

The core of the system is the character modules, one per character. In order to generate natural behaviour throughout, HCA will always be in one of three states. He is either (a) alone in his study, writing, reading, etc.; (b) receiving user input; or (c) producing output. (a) is handled by the non-communicative action (NCA) module; (b) is

handled by the communicative function (CF) module; and (c), HCA's communicative actions (CAs), are handled by the mind state agent (MSA). We need state (c) for HCA to process the input to respond appropriately. For believability, HCA's behaviour must show that he is aware that someone is addressing him (a) [2]. And even when not engaged in conversation, HCA is still observably present in his study. There, he should behave as humans would when on their own in their study, such as work or read (b). The HCA CM is controlled by the character module manager (CMM) which communicates with the other main modules and controls CM-internal communication. The CMM uses a conversational history to record all user input and system output and do general-purpose semantic book-keeping. HCA imposes his conversational goals agenda and personality on the dialogue through the communicative intention planner. Goals, input, and history information are handled by domain agents and an HCA knowledge base rather than by a hard-coded dialogue structure (interaction model) as in task-oriented systems. HCA's emotions are handled by an emotion grammar for modifying emotions as a function of goals, input, and discourse context.

4 Non-Communicative Action and Communicative Functions

When a user ends conversation, the system goes into the NCA state. The NCA module ensures that HCA performs contextually meaningful behaviour when not engaged in conversation. The design principles for HCA's repertoire of non-communicative actions are that (1) these should be consistent with his default emotional state, and (2) the non-communicative actions will be ultimately repetitive. By (1), we assume that HCA returns to his default emotional state after a conversation. HCA's NCA behaviour is ignorant of the emotions in the previous conversation and remains emotionally stable in the NCA state. (2) does not imply any important reduction in HCA's NCA behaviour. When humans work or read, for instance, their observable behaviour probably is limited and quasi-repetitive. HCA needs a repertoire of NCAs, so that users are unlikely to spot repetition, cf. Table 1. The table makes clear that NCA is fundamental to conversation. When HCA, e.g., listens to a user, he does so as determined by the NCA. The same is partly true when he produces conversational output.

Table 1. NCA repertoire: emotional background, action sequences, and behavioural elements

Behaviours Output modalities	Behavioural elements and action sequences
Speech	mumble unintelligibly, mumble words out of context
Other acoustic elements	hum a tune, grunt, hrmmm, action noises
Gaze	look down, look into space, roll eyes, stare, close one eye
Gesture	clasp hands front or back,, put hands in pockets, fold arms, one/two-arm gesture, wave arms, scratch head
Facial expression	look thoughtful, smile, set lips in various ways
Head movement	nod, shake head
Body posture	stretch body (sitting or standing)

Emotional background	friendly, at ease
Action sequences	write, walk around, look at books on shelf, sit in easy chair thinking or reading, look at wall pictures

The relationship between NCA, CF, and CA states is: [NCA -> CF <-> CA -> NCA]. NCA must be followed by a CF when a new user addresses HCA. After the user's first input, conversation [CF <-> CA] may iterate, eventually followed by the NCA state. The CF module handles HCA's behaviour during user input turns. Thus, if HCA is writing and is being addressed, he must acknowledge the fact by, e.g., stopping to write, looking up at the user, and smiling. He will only put down his pen if the user speaks/gestures a second time. Otherwise, he will resume his writing.

The design principles for HCA's repertoire of NCAs are that these should be (1) natural and varied but not exaggerated. We should avoid the stereotypical nod-cum-"yes" pattern observed in early animated characters [2]; (2) consistent with HCA's current emotional state as determined by his latest CA; (3) coordinated in the same way as human behaviour is coordinated, (4) real-time fast, i.e. not awaiting the result of the concurrent input processing, and (5) "poker faced" in the sense that, since HCA is not processing the input as fast as the CFs, he should not give this away. HCA's CF repertoire for the first NICE prototype is shown in Table 2. The CF randomises the behavioural element instructions to the response generator.

Table 2. CF repertoire: behavioural elements and graceful NCA dismantlement

Behaviours	Behavioural elements and action sequences
Output modalities	
Speech	yes, right, okay, ehmm, well, none
Gaze	look at user, look away, look down, stare, blink
Facial expression, body	determined by latest CA
Head movement	nod, none
Physical action	suspend NCA, dismantle NCA

Acknowledgement and references

The work described is supported by the EC's Human Language Technologies Programme, Grant IST-2001-35293. The support is gratefully acknowledged.

1. Bernsen, N.O., Dybkjær, H., Dybkjær, L.: Designing Interactive Speech Systems. From First Ideas to User Testing. Springer-Verlag, Berlin (1998)
2. Cassell, J., Sullivan, J., Prevost, S., Churchill, E. (eds.): Embodied Conversational Agents. MIT Press, Cambridge, MA (2000)