PUBLISHED IN *Computer Animation '97*, Geneva, Switzerland, June 5-6, 1997, pp. 134-143

Layered, Modular Action Control for Communicative Humanoids

Kristinn R. Thórisson

Gesture & Narrative Language Group[✈]
The Media Laboratory
Massachusetts Institute of Technology
20 Ames Street, Cambridge, Massachusetts 01239
kris@digi.lego.com http://www.media.mit.edu/~kris

Abstract. Face-to-face interaction between people is generally effortless and effective. We exchange glances, take turns speaking and make facial and manual gestures to achieve the goals of the dialogue. This paper describes an action composition and selection architecture of characters capable of full-duplex, real-time face-to-face interaction with a human. This architecture is part of a computational model of psychosocial dialogue skills, called *†mir*, that bridges between multimodal perception and multimodal action generation. To test the architecture, a prototype humanoid has been implemented, named *Gandalf*, who commands a graphical model of the solar system, and can engage in task-directed dialogue with people using speech, manual and facial gesture. Gandalf has been tested in interaction with users and has been shown capable of fluid turn-taking and multimodal dialogue. The primary focus in this paper will be on the action selection mechanisms and low-level composition of motor commands. An overview is also given of the †mir model and Gandalf's graphical representation.

1 Introduction

Endowing computers with an ability to engage in face-to-face interaction marks the beginning of a new era in our relationship with machines—one that relies on communication, social convention and dialogue skills. The work described in this paper is motivated by the idea of such communicative, autonomous agents. The interest is not merely in natural language—and surely there have been numerous projects on that are language-only [c.f. Chin 1991]—but rather a multimodal system, duplicating face-to-face dialogue between two or more communicating humans. Clearly this would rely on traditional computer graphics and natural language research, yet it goes beyond it in obvious ways, requiring input from artificial intelligence and psychological research as well.

Since the emphasis is on communication abilities that only humans are capable of, we user the terms "communicative humanoids". To this end, Cassell et al. [1994a, 1994b] describe a system for automatic speech and gesture generation. The system employs two graphical humanoid (human-like) characters that interact with each other using speech, gaze, intonation, head and manual gesture. The system employs what the authors call PaT-Nets (Parallel Transition Networks) in which synchronization between gestures and speech is accomplished as simultaneously executing finite state machines. The system provides an insight into the complexities of synchronizing various levels of multimodal action generation, from the phoneme level up to the phrase and full utterance. Perlin & Goldberg [1996] describe efforts toward similar goals using very different methods (and no emphasis on understanding and generating dialogue algorithmically). If we want these synthetic characters to comprehend and generate natural language [Allen 1987], gesture [McNeill 1992, Poyatos 1980], body movements [Goodwin 1986], facial gestures [Ekman & Friesen 1987], back channel feedback [Yngve 1971], speaking turns [Goodwin 1981], etc., and do this in real-time interaction with people, we need to make a number of additions, the primary being *perception*, mechanisms for *real-time control* and *knowledge bases for dialogue.*

The characteristics of embodied multimodal dialogue have been summarized in Thórisson [1996, 1995a]; of particular interest here are the following:

1. *Multi-layered Input Analysis and Output Generation*. In discourse, responses in one mode may overlap another in time, and constitute different information [Cassell & McNeill 1992, McNeill 1992, Goodwin 1981]. The overlaping actions can be anything from very short responses like glances and back channels, to tasks with longer time spans, such as whole utterances and topic continuity generation. In order for purposeful conversation to work, reactive and reflective[1] responses have to co-exist to provide for adequate behavior of an agent.
2. *Temporal Constraints.* Certain responses are

[✈] Now at LEGO a/s, Kløvermarken 120, 7190 Billund, Denmark expected to happen within a given time span, such as looking in a direction being pointed in [Goodwin 1981]. If these rules are violated, e.g. the direction of gaze changes 5 seconds after it was expected to change, the action's meaning may be drastically changed in the context of the dialogue.

3. *Functional & Morphological Substitutability*. Functional substitutability refers to the phenomemon when *identical looking acts can serve different dialogical functions.* Morphological substitutability is the reverse: *Different looking acts can serve the same function.*

Special features of animation and action control in the †mir architecture that will be dicussed in this paper can be summarized as the following:

- The action control scheme fits directly as the back-end of virtual characters with full-loop perception-action autonomy, capable of language understanding and generation.
- Motor actions are split into two phases; a *decision* (or intentional) phase and a *composition/execution* phase.
- Intentions to act vary in their *specificity:* the more specific an intention is (e.g. blinking) the fewer morphologies (ways to do it) exist; the less specific it is (e.g. "looking confused") the more options there are in the way it may eventually be realized.
- The *final morphology* ("form" or "look") of an intended action is *chosen at run-time.*

2 Related Architectures

Blackboard architectures, which can be traced back to Selfridge's Pandemonium system [Selfridge 1959], were invented as a method to handle unpredictable information like that encountered in speech recognition and planning [Nii 1989, Hayes-Roth et al. 1988]. The blackboard architecture attacks the problem of unpredictability by the use of a common data storage area, or blackboard, where results of intermediate processes, or knowledge sources, are posted and can be inspected by other processes working on the same problem. Modifications to the original HEARSAY blackboard architecture for speech recognition [Reddy et al. 1973] include mechanisms to allow interleaved execution of subsystems, as well as communication between them [Fehling et al. 1989], resource management, speed/effectiveness trade-off and reactive systems behavior [Dodhiawala 1989]. These additions are very useful for real-time systems.

Research on errors in human and animal locomotion has supported a model in which distinct levels of representation are at work for any motor act [Rosenbaum et al. 1992]: Levels activated earlier provide information spanning longer stretches of time, levels actuated later provide smaller and smaller constituents for that behavior. Rosenbaum et al. [1992] have proposed what they call the Knowledge Model: Motor control is performed by modules that carry information about *postures.*

The NASREM architecture [Albus et al. 1987] integrates results from research on animal sensory-motor skills into a comprehensive scheme for autonomous-robot and telerobot control. The system contains multiple levels of processing, each level containing the three components of sensory processing, world modeling and task decomposition. A global data storage (blackboard) is accessible from any level.

Working on creating insect-like robots, Brooks [1990] proposed what he calls a *Subsumption* architecture where low-level behaviors of a robotic agent can be subsumed by higher-level, later-designed behaviors. This allows for incremental development of robot skills and a robustness that is difficult to achieve with traditional methods. Another behavior-based architecture are Maes' *competence modules*—software modules that contain enough information to execute a particular behavior from beginning to end [Blumberg 1996, Maes 1989]. The modules are connected together by activation links that control their sequence of execution. The input to the modules can come both from internal goals and the environment. This architecture, and other related approaches [Wilson 1991, Steels 1990, Agre & Chapman 1987] are very good for effective, fast action selection, and some allow learning. However, they lack methods to deal with external and internal time-constraints and are limited in the planning they can handle.

2.1 Problems

The problem with all of the reviewed systems is a lack of one of the following three crucial ingredients in all face-to-face dialogue: {1} Multimodal action generation, {2} use of natural language, {3} real-time response. There exists a strong dichotomy in most of the systems between language capability and action generation/coordination; only Cassell et al.'s system [1995a, 1995b] integrates both in a consistent way. The Blackboard architecture is a general proposal to deal with ill-defined problem solving—its adoptation to multimodal systems left unspecified. An obvious problem with behavior-based systems such as Brooks' [1990] and Maes' [1989] is that because the interfaces between action control modules are defined at a relatively low level, creating

[1] Intuitively, the terms *reactive* and *reflective* refer to fast and slow responses, respectively. A more specific definition can be found in Thórisson [1996a, 1995a].

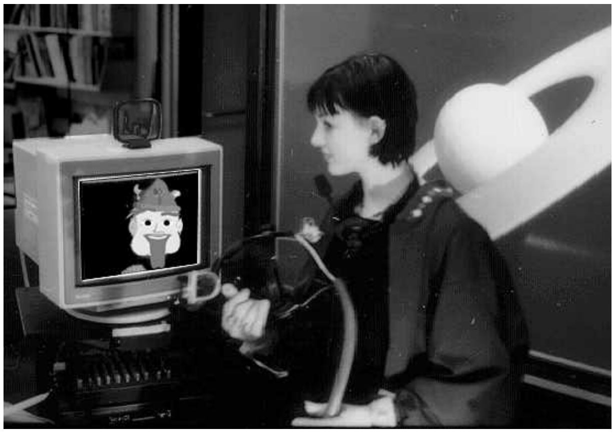

FIGURE 1. The prototype setup includes a small monitor for Gandalf's face and hand, and large monitor for the topic of discussion. A user can ask Gandalf to travel to the planets in the solar system, and ask questions about each one. The user in this picture is wearing a body-tracking suit and holding the eye tracker.

large systems in them can be problematic at best, impossible at worst. Few environments are more real-time than face-to-face dialogue, but dialogue is based on natural language. By far the greatest problem with behavior-based systems is adding natural language capabilities to them, requiring extensions the architectures are ill-suited for.

While several ideas from these above systems are applicable to the problem of real-time, multimodal, face-to-face dialogue, no single architecture provides a complete solution. What is called for is a new architecture capable of integrating all the necessary elements to support and sustain multimodal dialogue with people.

3 †mir & Gandalf: Overview

†mir[2] is a computational, generative model of psychosocial dialogue skills which can used to create computer-driven characters capable of multimodal perception and action generation [Thórisson 1996a]. It borrows several features from the above blackboard and behavior-based artificial intelligence (A.I.) architectures, but goes beyond these in the amount of communication modalities (speech, intonation, body language, facial & manual gesture) and performance criteria it addresses. To provide necessary context for the discussion on layered action control that follows, we will first give a short overview of the †mir architecture.

3.1 Layers

†mir contains three types of processing modules: *perceptual*, *decision* and *behavior*. The modules are found three layers, {1} a Reactive layer (RL), {2} a Process Control layer (PCL), and {3} a Content layer (CL), and a module called the Action Scheduler (AS). Each of the three layers contain perception and decision modules; perception modules with specific processing demands provide the necessary information about the state of the world to support decisions about behaviors with a specific perceive-act cycle time. Decisions to act resulting from processes in the RL generally have response cycles under 1 second, typically in the 150-500 ms range—actions like blinking and determining the next fixation point. Decisions in the PCL have a frequency around 1 Hz and up—actions like taking speaking turn or looking at someone who is addressing you. Processing in the CL has response times from seconds up to infinity; the CL contains the topic knowledge of an agent, in the form of one or more knowledge bases. †mir makes a clear distinction between *dialogue knowledge* and *topic knowledge*: the former relates to general issues of dialogue management such as watching manual and facial gesture of a speaker, following her gaze and taking turns, the latter relates to specific issues of the topic such as the orbit of the planets in the solar system, which clearly have nothing to do with the dialogue process. This has the added benefit of allowing the addtion of knowledge-bases in a modular fashion.

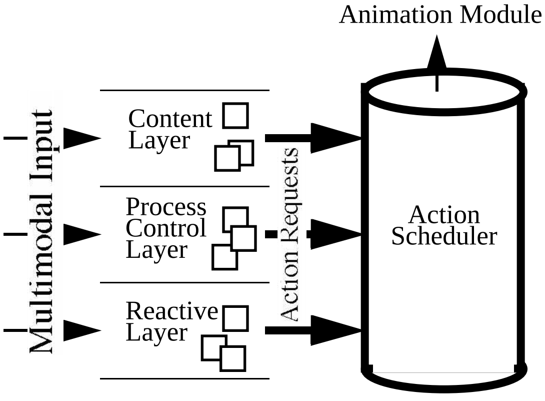

FIGURE 2. Simplified overview of the †mir architecture. Any multimodal input data generated by a real human can map into any of the layers; perceptual modules in each layer can access each other's results through semi-global blackboards. Decision modules (small squares) operate on these results and decide when to send Action Requests to the Action Scheduler. Not shown are the systems perceptual system, its blackboards and knowledge bases.

[2] Pronounced *e-mir*, with the accent on the first syllable. The name comes from Nordic mythology.

3.2 Action Selection & Composition

In †mir, actions get selected in a two-stage process. First, a particular decision module fires because sufficient conditions in corresponding perceptual module(s) have been fulfilled (see Figure 2). Once a decision module fires, it creates an *action request*. This request can be considered a “potential to act” (all events at this point in the system are non-deterministic). The fate of the request is determined in the next stage, the Action Scheduler.

Before action requests are turned into visible behavior, they are sent to the Action Scheduler (AS), which composes the exact morphology of an action. It prioritizes them, and decides how each requested action should look at the lowest (motor) level, according to the current status of the motor system (the agent’s face and body). We will now give an overview of the prototype hardware setup.

3.3 Prototype

A prototype agent called Gandalf has been implemented in †mir, which runs on 8 networked computers (2 DECs, 2 SGIs, 2 PCs, 1 HP, 1 Mac). To capture the user’s multimodal actions, speech, prosody, gaze & body movements, it uses a microphone (the agent's "ear" [Thórisson 1996a, BBN 1993]), an eye tracker and a body suit (the agent's "eyes" [Bers 1996]). Gandalf’s behaviors were based on a through review of the psychological literature on face-to-face conversation [c.f. Goodwin 1981, Sacks et al. 1974]. The prototype has been tested in interaction with naive users and proven to be capable of fluid turn-taking and unscripted dialogue [Thórisson 1996a, Thórisson & Cassell 1996]. A short overview of the full system can be found in Thórisson [1997, 1995b]; a complete description is provided in Thórisson [1996a]. For the remainder of this paper we will concern ourselves with the philosophy behind the action control mechanism in †mir, it’s current implementation, and the low-level graphical engine that animates the prototype character.

4 Action Scheduling & Composition: Motor Composer & Behavior Lexicon

When decisions to act—move the creature’s muscles—are generated in semi-autonomous layers, an arbitration has to take place to avoid conflicts in the access to these “motors” (or muscles, pulleys, degrees of freedom). A character’s body and its associated motors become therefore a resource that has to be managed. In †mir, this management is handled by the Action Scheduler (AS). The approach taken in the AS is in some ways similar to Rosenbaum et al.’s [1992] approach to motion control. Their idea of stored postures is used in the Action Scheduler, as is the idea of hierarchical storage of increasingly smaller units.

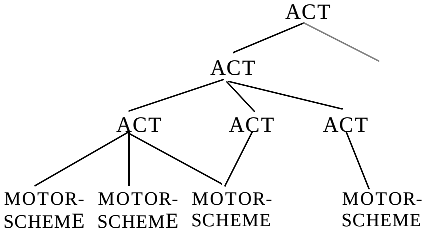

FIGURE 3. Acts and motor-schemes create a hierarchy of continuous abstraction that can be entered at any level, from each of the layers in the system. The motor-schemes contain a list of the motors to move, and their absolute positions.

The AS consists of two basic elements: A *Behavior Lexicon* and a *Motor Composer*. The Lexicon is a network of *behaviors* defined as a hierarchy (Figure 6); the Motor Composer is an anytime algorithm[3] that selects the final morphology of a behavior based on {1} the agent’s body state, {2} which layer initiated the action request, and {3} the age of the action request.

4.1 Behavior Lexicon

The Behavior Lexicon, the other main component of the AS, contains behaviors organized in an abstraction hierarchy, specific motor programs in the leafs:

1. ACT behaviors and
2. MOTOR-SCHEME behaviors (leaf nodes).

Each leaf node has a mapping to a particular motor configuration (either dynamic or static, e.g. a smile or a blink), by listing the specific motors and motor sequences needed. ACTs, the nodes above the leafs, describe behaviors that can be realized in more one way.

ACT behaviors contain two slots: [1] A NAME—the behavior’s unique name, and [2] OPTIONS—a list of alternative behaviors that can be used to satisfice the action request; each option is a list of behaviors, which is a list of the form [NAME, EXEC-TIME, DEALY], where NAME is the behavior’s name, EXEC-TIME is the execution time for that behavior, and DELAY is a time-delay that offsets this behavior’s execution from the execution of the behavior that subsumes it. To explain, normally all motor

[3] Anytime algorithms improve their output linearly over time and can always be interrupted for a partial solution [Dean 1987].

```
(setf *Behavior-Lexicon*
  ;ACT TEMPLATE:
   ;(name class (((act-name-of-option-1 delay
                                        exec-time)
   ;              (act-name delay exec-time) etc*)
   ;              (etc*)))
  ;MOTOR-SCHEME TEMPLATE:
   ;(motor-name class delay exec-time rel-pos)
   '(
     ; MORPHOLOGICAL DEFINITIONS
        ;Features
          ;neutral
     (face-neutral act
           (((mouth-neutral 100 400)
             (eyes-neutral 0 300)
             (brows-neutral 0 500))))
     (brows-neutral act
           (((left-brow-neutral 0 400)
             (right-brow-neutral 0400))))
     (left-brow-neutral motor-scheme
           (((Bll 0 400 30)
             (Blc 0 400 30) ;Brow, left, central
             (Blm 0 400 30)))) ;Brow, left, medial
     (right-brow-neutral motor-scheme
           (((Brm 0 400 30)
             (Brc 0 400 30)
             (Brl 0 400 30))))
     ;FUNCTIONAL DEFINITIONS
     (blink act (((close-eyes 0 50)
                  (open-eyes 50 50))))
     (close-eyes motor-scheme (((Eru 0 300 10)
                                (Elu 0 300 10))))
     (open-eyes motor-scheme (((Eru 0 300 75)
                               (Elu 0 300 80))))
     . . . .
```

FIGURE 4. A fragment of the Behavior Lexicon for the Gandalf prototype. (Figure 8 shows the names of the facial motors.) Gandalf's Behavior Lexicon contains a total of 83 behavior nodes.

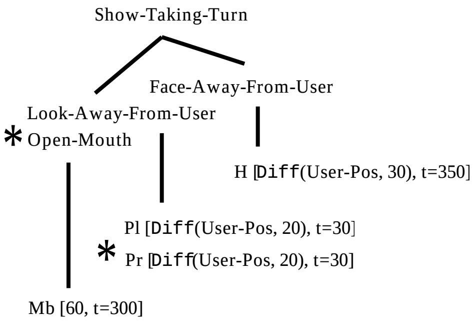
Show-Taking-Turn
Face-Away-From-User
Look-Away-From-User
*
Open-Mouth
H [Diff(User-Pos, 30), t=350]
Pl [Diff(User-Pos, 20), t=30]
*
Pr [Diff(User-Pos, 20), t=30]
Mb [60, t=300]

FIGURE 5. A mockup example of a section of the Behavior Lexicon. The behavior SHOW-TAKING-TURN has two possible instantiations, TURN-AWAY-FROM-USER (turn head) and the parallel pair {LOOK-AWAY-FROM-USER, OPEN-MOUTH}. Each of these point to low-level motor commands (motor schemes) with degrees (for rotating eyes and head) and time (in milliseconds). The function Diff returns an orientation that is guaranteed to not include the user's position in the agent's line of sight. Parallel actions are marked with a star.

H = agent's head, P = pupil (left and right), t = time in milliseconds, other numbers represent degrees (and relative position in the case of motor Mb), Mb = bottom mouth motor.

specifications of a behavior get sent to the animation unit at the same time. An example of such a behavior is the behavior SMILE: corners of the mouth are moved upward and outward, while the lower eyelids are moved slightly up from their resting position, all at once. For sequential actions, certain motor commands may have to wait for others to finish. The DELAY of a motor determines how long after the whole action started it should begin execution. An example of a behavior that uses this feature is the behavior BLINK: first the eye is closed, then opened.

Each MOTOR-SCHEME behavior has a [1] NAME slot—the behavior's unique name, and [2] MOTOR-LIST—a list of the motors involved. Each motor in this list contains [1] MOTOR-NAME—the motor's unique name, [2] EXEC-TIME—its default execution time, and [3] REL-POS—the motor's goal position, relative to its range of motion.

ACT behaviors can contain different execution times from the behaviors they subsume. To take an example, when the behavior EYES-NEUTRAL is executed as part of the higher-up behavior FACE-NEUTRAL, it may take 400 ms for the motors to get to their final position, but when EYES-NEUTRAL is called directly it may take only 100 ms. When execution times differ for each motor within the same behavior, the longest motor takes the *maximum* execution time to reach its goal (400 ms in this example), while the others become a percentage of that maximum. Using this scheme, a behavior's EXEC-TIME can be recalculated on the fly when the request is in the composition stage, in consideration of current time-constraints.

4.1.1 Function, Morphology

To address the morphological and functional substitutability of dialogue actions described in the introduction, behaviors come into two main classes: {1} *Morphological* and {2} *Functional*. Morphological behaviors are named after the way they *look*, for example, the behavior BROWS-IN-U-SHAPE specifies a shape for the brows to take. Nothing is said about what circumstances such a behavior should or could be used in, nor what possible meanings such a behavior could carry.

On the other hand, the behavior SHOW-TAKING-TURN specifies a dialogue *function*. There are many ways for showing that you are taking the turn to say something, one being opening the mouth slightly, another glancing away briefly [Kleinke 1986, Goodwin 1981, Duncan 1972]. This way, decision modules in each layer can issue requests for behaviors based both on function and look, which makes the system more powerful, and module construction easier.

4.1.2 Spatio-Motor Skills

To allow an agent to move in relation to surrounding objects such as a person or a task area, the AS needs access to a spatial knowledge base. Examples of such actions would be LOOK-AT-USER and TURN-TO-AREA-[X][4]. This is done with access to a common spatial knowledge base that is fed with information from the sensors (Figure 6).

4.2 The Motor Composer

In the current implementation, action requests are received from the layers in the form [ACTION-NAME TIME-STAMP EXPECTED-LIFETIME WHO], where WHO is one of RL, PCL or CL, TIME-STAMP is the time when the decision to act was made, and EXPECTED-LIFETIME is the pre-computed[5] lifetime of the request, the time beyond which the requested action is probably no longer relevant in the dialogue context.

4.2.3 Prioritization

As control of an agent's body is competed for by conflicting commands from different layers in the system, they have to be prioritized in some way. At the coarsest level, the Motor Composer prioritizes decisions in the following way: Decisions to act that were initiated by the RL (e.g. a decision to blink) are serviced immediately, those initiated by the PCL (e.g. to utter a sentence) take second priority and those initiated in the CL (e.g. to change the topic of the dialogue) take lowest priority. To prioritize further, the amount of overlap the behavior has with currently executing motors in the agent's body is used.

4.2.4 Action Timeliness

As mentioned before, real-time response is crucial in face-to-face communication, and ensuring the timely delivery of communicative behaviors is key to making the interaction work. Intentions to act in †mir are ensured timeliness two ways: {1} by the priority scheduling (discussed above) and {2} by a time-management system that ensures that actions that didn't get executed in time will not be. If the expected lifetime has been reached, and no MOTOR-SCHEME behavior has been found for it yet, the action is cancelled:

Cancel [A] IF (TIME-NOW > TIME-STAMP$_A$ + EL$_A$)

where A is the action requested and EL$_A$ is the expected lifetime of that action.

4.3 A Trace of AS Processing

The following steps summarize the events relating to the path of a single action request in the Action Scheduler, from its reception to its execution:

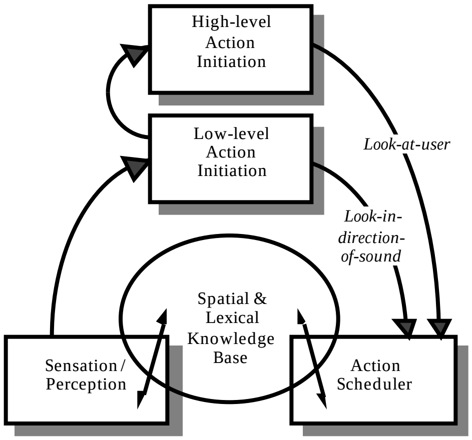
High-level Action Initiation
Low-level Action Initiation
Look-at-user
Look-in-direction-of-sound
Spatial & Lexical Knowledge Base
Sensation / Perception
Action Scheduler

FIGURE 6. The Action Scheduler has access to a spatial knowledge base that is kept updated by the sensory and perceptual mechanisms. Examples of messages sent to the AS from the Reactive and Process Control layers are shown in italic letters.

1. An *action request*, A_R, is received from one of the three layers (e.g. a "blink" request from the RL).
2. The request is prioritized according to which layer it came from.
3. Once the A_R's turn comes, check its expected lifetime, E_L. If it has been reached, cancel the request and go on to the next A_R. Otherwise,
4. ...it's associated action (e.g. "blink") is found in the Behavior Lexicon and a particular motor scheme for it, M, is retrieved.
5. If there are no more motor schemes available for the request, or, if the request's E_L has been reached, execute M and service next Action Request. Otherwise, if there are more motor schemes available,
6. the current status of the motors needed for the particular motor scheme (e.g. eyelids) is used to give motor scheme M a score. The score depends on how many of the motors required for M are currently busy executing another action.
7. Then we go back to step 3 and retrieve M'. We also give M' a score according to step 5.
8. Compare the scores of M and M'. Keep the M with the better score (because it will create less conflicts with the currently executing behaviors).
9. Continue the process from step 4 until a motor scheme has been executed.

An important point in step 3 is that the information about the effect of not being able to execute an action in time is expected to flow back through the character's sensors as a particular reaction of the user to the lack of behavior. Since there is a tight loop of perception going in to the

[4] This behavior, unlike the other examples, contains a variable. For complex behaviors, variables are essential.

[5] This value could (and in many cases should) be computed dynamically. This increases the need for "bookkeeping" in the system and thus its complexity.

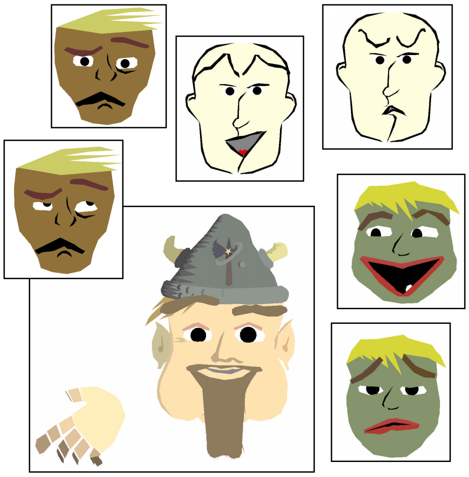

FIGURE 7. Faces and facial expressions generated with the ToonFace facial representation. The largest face is Gandalf's.

agent, any problem in the global aspects of dialogue should show up there instantaneously and a new action would be triggered. Thus, the need for complex bookkeeping protocols *within* the system—as opposed to through the *outer* feedback loop by way of the effect the agent's behavior has on the user's behavior—should be diminished, if not eliminated.

Once a final motor scheme has been selected, the individual motor commands are sent to an animation module that executes multi-threaded motor commands. We will now take a look at this module, and the representation used for the agent.

Rather than building dependencies between behaviors into the Behavior Lexicon (i.e. if "blink-slowly" fails to execute, execute "blink-fast" instead), a concept called cascaded decision modules is used. By cascading a number of decision modules, corresponding to a number of behavior morphologies, each triggered in the case of another's cancellation, inappropriateness or failure, whole classes of behaviors can be built up.

4.4 Ballistic Execution

When, in the process from decision to act to the act's execution, does the action, or part of the action, become impossible to cancel? This question about when to go ballistic is an important one. As discussed in Thórisson [1995a] the incremental and reactive nature of dialogue allows participants to interrupt each other at a moment's notice. In †mir, any commands leaving the AS are ballistic. This last part of the path should therefore be kept very short, typically less than a second. For actions longer than a second, one would expect them to be composed—or at least *executed* [Kosslyn & Koenig 1992]—incrementally so that they can be cancelled at any time. In the current prototype, the AS takes care of segmenting sequential actions: any motor command with a relative offset from the others in a motor-scheme will be witheld and only sent to the AS when the appropriate time has lapsed. This makes the interruption of sequential behaviors straight forward.

Just as motor commands are ballistic once they leave the AS, speech leaving the AS is also ballistic. (The content of speech is generated in the knowledge bases, except for a few reactive verbal acts like saying "ahhh" when hesitating and giving back channel feedback [Yngve 1970]). It is therefore important that the speech is segmented correctly to allow for cancellations in case the user interrupts the agent. Thus, with an utterance of 10 words only 3-4 words would be committed to ballistic execution at a time, the others witheld in case there is an interruption and the character needs to shut up. Currently, the segmentation is done at natural boundaries larger than the word but shorter than the sentence. Noun phrases, verb phrases and fillers are all sub-components that give useful (albeit not *always* appropriate) boundaries. How the AS controls the incremental execution of long actions, and the specifics of its communication with the knowledge bases remains an issue of further research.

5 Face / Hand Representation & Animation

5.1 Face

Making facial computer animation look convincing has proven to be a difficult task. Most current systems for facial animation are very complex, include between 70 and 80 control parameters, require powerful computers and seldom run in real-time [c.f. Pelachaud et al. 1996, Ekman & Friesen 1978]. An alternative is what might be called a "caricature" approach where important features are exaggerated. ToonFace [Thórisson 1996b] is an attempt to create such an animation package. The primary goal of ToonFace is to create facial expressions in real time in response to a human interacting with it. ToonFace meets this requirement by being simple: A face is represented as 2-dimensional polygons and polygon groups with control points that can be manipulated in one or two dimensions. Figure 8 shows the control points. Although this representation only has 21 df, it is surprisingly expressive (see Figure 7). Each control point can move through a range, subdivided into 100 steps. A software package called ToonFace Editor allows the design of faces, the placement of the control points and their associated ranges of motion.

5.2 Manual Gesture

As mentioned before, †mir makes a clear distinction between dialogue knowledge and topic knowledge. This distinction separates manual gesture into two generation mechanisms. Four classes of dialogue-related manual gesture [Rimé & Schiaratura 1991] are independent of the topic knowledge base(s) used, and generated from dialogue knowledge: {1} *emblem gestures* related to the dialogue (e.g. holding up a hand to signal "Stop speaking!"), {2} *deictic gestures* (pointing to objects), {3} *beats* and {4} *butterworths*. These gestures are all initiated in the RL and PCL by requesting the appropriate type of gesture and providing the optional parameters (such as a 3-D vector or posture for deictic gestures) and treated in the same way as other actions in the Action Scheduler. Since iconic, pantomimic and deictic gestures related to the topic of discussion cannot be generated without reference to knowledge of the topic, and the knowledge residing in the dialogue system contains no topic knowledge, these are generated in the corresponding knowledge base, contained in the Content Layer. McNeill [1992] and Cassell et al. [1994a, 1994b] have proposed that beat gestures are generated by a system in a speaker's mind that is separate from other manual gesture generation mechanisms; †mir goes a step further and claims a distinction between topic-related gesture and process-related gesture. Whether this distinction has any correspondence in the way people generate manual gestures remains to be tested empirically.

The hand is currently animated by representing separately the hand's *position* and *shape* [c.f. Wexelblatt 1994, Sparrell 1993], and by giving the hand two states, *at-rest* and *active*. Whenever the animation module receives a command for a manual gesture it will execute the given type of gesture for the requested time period, after which it moves it back to its *at-rest* position. The gestures also have some controllable parameters, such as a *pitch* and *yaw* for deictic gestures, and *duration* for beat gestures. Gestural interruptability has been implemented: If a gesture is executing when a new hand gesture command arrives, the current action will be cancelled, and the new command will take over. The shape of the hand is interpolated from its current state to the shape associated with the first position in the new gesture, while the hand is moved linearly from its current position to the first position of the new command.[6] This scheme looks surprisingly natural considering its simplicity.

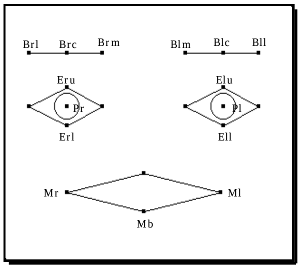

FIGURE 8. Movable control points—or motors—are coded as shown: Bll = brow, left, lateral; Blc = brow, left, central; Blm = brow, left, medial; Elu = eye, left, upper; Ell = eye, left, lower; Pl = pupil, left; Ml = mouth, left; Mr = mouth, right; Mb = mouth, bottom. Brow, pupil and eye are mirrored on the right side of the face. Head motion is coded as H. All motors are referenced with an absolute position from 0 to 100. Motors with two degrees of freedom are addressed by either h or v, for horizontal and vertical motion, respectively. All motors can be addressed and run in parallel.

5.3 Animation Engine

The animation unit provides the agent with muscles. It receives commands from Action Scheduler in the form [MOTOR, POSITION, TIME], where MOTOR is the motor to move (see Figure 8), POSITION is the new (absolute) position it should move to and TIME is the absolute time it should take to get there. Except for manual gesture, the commands received by this unit are all ballistic.

The current prototype for †mir uses the ToonFace Animator [Thórisson 1996b] as the animation engine, which runs on an SGI Indigo$_2$. The loop time for a complete redraw of the face and hand is currently 150 ms, which is, because of computational limitations, somewhat higher than the cycle time of 50 ms sufficient for most behavior encountered in multimodal communication [Thórisson 1996a, 1995a, 1993].

6 Summary & Future Work

The action/animation control mechanism described in this paper results in a system with, among other novel features, the following unique general characteristics:

- Motor actions are split into two phases; a *decision* (or intentional) phase and a *composition/execution* phase.
- Behaviors are represented as a hierarchical knowledge base for actions, where actions contain *postures* with associated *destination travel times.*
- Behaviors are defined both *morphologically* and *functionally*.
- The *final morphology* ("form" or "look") of an intended action is *chosen at run-time.*

[6] Thanks to Hannes Vilhjálmsson for his contributions to and implementation of the hand and the gesturing mechanism.

- Modules for both reactive and reflective behaviors can be added incrementally, and the behavior repertoire can be extended without needing a redesign of the behavior lexicon.

The above characteristics lead to high modularity and the possibility of incremental design, as well as increasing a creature's responsiveness to its surroundings and real-time events. Furthermore, the action control scheme fits directly as the back-end of virtual creatures with full-loop perception-action autonomy, capable of language understanding and generation. The features specific to human-like communication skills are:

- A character's behavior is interruptible at natural points in its interaction with its environment, without being rigid or step-lock.
- Gesture and facial expression are an integrated part of the communication, with no artificial communication protocols.
- Concurrent behaviors, such as glancing over to an object the speaker points at, happen naturally, at the right times, and where they are expected.
- When speech overlaps or miscommunication occurs, it is dealt with in the same ways as in human face-to-face interaction, by stopping, restarting, hesitating, etc.

Future work focuses on extending the architecture to handle navigation and object manipulation. Also we want to extend Gandalf's knowledge of dialogue, as well as topic. The final result of this research is a very interactive, seemingly intelligent character that is fun to talk to.

Acknowledgments. This work was sponsored by Thomson-CSF, the M.I.T. Media Laboratory, the TSG Foundation and RANNÍS. The author would like to thank Justine Cassell, Pattie Maes, Steve Whittaker & Richard A. Bolt for guidance; Joshua Bers, David Berger, Christopher Wren, Steven Levis, Calvin Yuen, Nimrod Warshawsky, Roland Paul & Hannes Vilhjálmsson for technical assistance and Jennifer Glos, Scott Prevost & Marina Umachi for useful suggestions.

References.

Agre, P. E. & Chapman, D. (1990). What Are Plans for? In P. Maes (ed.), *Designing Autonomous Agents*, 17-34. Cambridge, MA: MIT Press.

Albus, J, McCain, H & Lumia, R. (1987). NASA/NBS Standard Reference Model for Telerobot Control System Architecture (NASREM), Technical Report Technical Note 1235, National Bureau of Standards, Gaithersburg, Maryland.

Allen, J. (1987). *Natural Language Understanding*. Reading, MA: Benjamin/Cummings Publishing Co. Inc.

BBN (1993). HARK Recognizer Release 1.1 Beta. Document No. 100-1.1. Bolt, Beranek & Newman, Inc., Speech and Natural Language Processing Department.

Bers, J. (1996). A Body Model Server for Human Motion Capture and Representation. *Presence: Teleoperators and Virtual Environments*, 5(3).

Blumberg, B. (1996). Old Tricks, New Dogs: Ethology and Interactive Creatures. Ph.D. Thesis, Massachusetts Institute of Technology.

Brooks, R. (1990). Elephants Don't Play Chess. In P. Maes (ed.), *Designing Autonomous Agents*, 3-15. Cambridge, MA: MIT Press.

Cassell, J., Pelachaud, C., Badler, N., Steedman, M., Achorn, B., Becket, T., Deouville, B., Prevost, S. & Stone, M. (1994a). Animated Conversation: Rule-based Generation of Faceial Expression, Gesture & Spoken Intonation for Multiple Conversational Agents. *Proceedings of SIGGRAPH '94.*

Cassell, J., Stone, M., Douville, B., Prevost, S., Achorn, B., Steedman, M., Badler, N. & Pelachaud, C. (1994b). Modeling the Interaction between Speech and Gesture. *Sixteenth Annual Conference of the Cognitive Science Society*, Atlanta, Georgia, August 13-16, 153-158.

Cassell, J. & McNeill, D. (1991). Gesture and the Poetics of Prose. *Poetics Today*, 12(3), Fall, 375-404.

Chin, D. N. (1991). Intelligent Interfaces as Agents. In J. W. Sullivan & S. W. Tyler (eds.), *Intelligent User Interfaces*, 177-206. New York, NY: Addison-Wesley Publishing Company.

Dean, T. L. (1987). Intractability and Time-Dependent Planning. *Proceedings of the 1986 Workshop on Reasoning About Actions and Plans*, M. P. Georgeff & A. L. Lansky (eds.), Los Altos, California: Morgan Kaufman.

Dodhiawala, R. T. (1989). Blackboard Systems in Real-Time Problem Solving. In Jagannathan, V., Dodhiawala, R. & Baum, L. S. (eds.), *Blackboard Architectures and Applications*, 181-191. Boston: Academic Press, Inc.

Duncan, S. Jr. (1972). Some Signals and Rules for Taking Speaking Turns in Conversations. *Journal of Personality and Social Psychology*, 23(2), 283-292.

Ekman, P. & Friesen, W. (1978). *Facial Action Coding System*. Palo Alto, CA: Consulting Psychologists Press.

Fehling, M. R., Altman, A. M. & Wilber, B. M. (1989). The Heuristic Control Virtual Machine: An Implementation of the Schemer Computational Model of Reflective, Real-Time Problem-Solving. In Jagannathan, R. Dodhiawala & L. S. Buam, *Blackboard Architectures and Applications*, 191-218. Boston: Academic Press, Inc.

Goodwin, C. (1986). Gestures as a Resource for the Organization of Mutual Orientation. *Semiotica*, 62(1/2), 29-49.

Goodwin, C. (1981). *Conversational Organization: Interaction Between Speakers and Hearers*. New York, NY: Academic Press.

Hayes-Roth, B., Hayes-Roth, F., Rosenschein, S. & Cammarata, S. (1988). Modeling Planning as an Incremental, Opportunistic Process. In R. Engelmore & T. Morgan, *Blackboard Systems*, 231-245. Reading, MA: Addison-Wesley Publishing Co.

Kleinke, C. (1986). Gaze and Eye Contact: A Research Review. *Psychological Bulletin*, 100(1), 78-100.

Kosslyn, S. M. & Koenig, O. (1992). *Wet Mind: The New Cognitive Neuroscience*. New York, New York: The Free Press.

Maes, P. (1989). How to Do the Right Thing. A.I. Memo No. 1180, December, Artificial Intelligence Laboratory, Massachusetts Institute of Technology.

McNeill, D. (1992). *Hand and Mind: What Gestures Reveal about Thought*. Chicago, IL: University of Chicago Press.

Nii, P. (1989). Blackboard Systems. In A. Barr, P. R. Cohen & E. A. Feigenbaum (eds.), *The Handbook of Artificial Intelligence*, Vol. IV, 1-74. Reading, MA: Addison-Wesley Publishing Co.

Pelachaud, C., Badler, N. I. & Steedman, M. (1996). Generating Facial Expressions for Speech. *Cognitive Science*, 20 (1), 1-46.

Perlin, K. & Goldberg, A. (1996). Improv: A System for Scripting Interactive Actors in Virtual Worlds. *ACM SIGGRAPH '96*, New Orleans, Louisiana.

Pierrehumbert, J. & Hirschberg, J. (1990). The Meaning of Intonational Contours in the Interpretation of Discourse. In P. R. Cohen, J. Morgan & M. E. Pollack (eds.), *Intentions in Communication.* Cambridge: MIT Press.

Prevost, S. (1996). A Semantics of Contrast and Information Structure for Specifying Intonation in Spoken Language Generation. Ph.D. Thesis, Faculty of Coputer and Information Science, University of Pennsylvania.

Poyatos, F. (1980). Interactive Functions and Limitations of Verbal and Nonverbal Behaviors in Natural Conversation. *Semiotica*, 30-3/4, 211-244.

Reddy, Y. V., Erman, L. D. & Neely, R. B. (1973). A Model and a System for Machine Recognition of Speech. *IEEE Transactions on Audio and Electro-Acoustics*, AU-21, 229-238.

Rimé, B. & Schiaratura, L. (1991). Gesture and Speech. In R. S. Feldman & B. Rimé, *Fundamentals of Nonverbal Behavior*, 239-281. New York: Press Syndicate of the University of Cambridge.

Rosenbaum, D. A. & Kirst, H. (1992). Antecedents of Action. In H. Heuer & S. W. Keele (eds.), *Handbook of Motor Skills*. New York, NY: Academic Press.

Sacks, H., Schegloff, E. A.. & Jefferson, G. A. (1974). A Simplest Systematics for the Organization of Turn-Taking in Conversation. *Language*, 50, 696-735.

Selfridge, O. (1959). Pandemonium: A Paradigm for Learning. *Proceedings of Symposium on the Mechanization of Thought Processes*, 511-29.

Sparrell, C. J. (1993). Coverbal Iconic Gesture in Human-Computer Interaction. Master's Thesis, Massachusetts Institute of Technology. Cambridge, Massachusetts.

Steels, L. (1990). Cooperation Between Distributed Agents Through Self-Organization. In Y. Demazeau & J. P. Müller (eds.), *Decentralized A. I.* Amsterdam: Elsevier Science Publishers B. V. (North-Holland).

Thórisson, K. R. (1997). Gandalf: A Communicative Humanoid Capable of Real-Time Mutlimodal Dialogue with People. *ACM First Conference on Autonomous Agents*, Marina del Rey, California, February 5-8.

Thórisson, K. R. (1996a). Communicative Humanoids: A Computational Model of Psychosocial Dialogue Skills. Ph.D. Thesis, Massachusetts Institute of Technology.

Thórisson, K. R. (1996b). ToonFace: A System for Creating and Animating Interactive Cartoon Faces. M.I.T. Media Laboratory Learning and Common Sense Section technical report 96-01, 13 pp.

Thórisson, K. R. (1995a). Computational Characteristics of Multimodal Dialouge. *AAAI Fall Symposium Series on Embodied Lanugage and Action*, November 10-12, Massachusetts Institute of Technology, Cambridge, 102-108.

Thórisson, K. R. (1995b). Multimodal Interaction with Humanoid Computer Characters. *Conference on Life-like Computer Characters*, Snowbird, Utah, September 26-29, p. 45 (Abstract).

Thórisson, K. R. (1994). Face-to-Face Communication with Computer Agents. *AAAI Spring Symposium on Believable Agents Working Notes*, Stanford University, California, March 19-20, 86-90.

Thórisson, K. R. (1993). Dialogue Control in Social Interface Agents. *InterCHI Adjunct Proceedings '93*, Amsterdam, April, 139-140.

Wilson, S. W. (1991). The Animat Path to AI. In J-A. Meyer & S. W. Wilson (eds.), *From Animals to Animats*. Cambridge, MA: MIT Press.

Wexelblatt, A. (1994). A Feature-Based Approach to Continuous-Gestures Analysis. Master's Thesis, Media Arts and Sciences, Massachusetts Institute of Technology.

Yngve, V. H. (1970). On Getting a Word in Edgewise. *Papers from the Sixth Regional Meeting.*, Chicago Linguistics Society, 567-78.