CHAPTER 12

A Distributed Architecture for Real-time Dialogue and On-task Learning of Efficient Co-operative Turn-taking

Gudny Ragna Jonsdottir and Kristinn R. Thórisson

1. Introduction

Building automatic dialogue systems that match human flexibility and reactivity has proven difficult. Many factors impede the progress of such systems, useful as they may be, from the low-level of real-time audio signal analysis and noise filtering to medium-level turn-taking cues and control signals, all the way up to high-level dialogue intent and content-related interpretation. Of these, we have focussed on the dynamics of turn-taking—the real-time[1] control of who has the turn and how turns are exchanged and how to integrate these in an expandable architecture for dialogue generation and control. Manual categorization of silences, prosody and other candidate turn-giving signals, or analysis of corpora to produce static decision trees for this purpose cannot address the high within- and between-individual variability observed in natural interaction. As an alternative, we have

[1] By "real-time" here we mean conducting dialogue at a pace acceptable to, and in line with, human expectations, as understood and learned from real-world experience.

developed an architecture with integrated machine learning, allowing the system to automatically acquire proper turn-taking behavior. The system learns cooperative ("polite") turn-taking in real-time by talking to humans via Skype. Results show performance to be close to that of human, as found in naturally occurring dialogue, with 20% of the turn transitions taking place in under 300 milliseconds (msecs) and 50% under 500 msecs. Key contributions of this work are the methods for constructing more capable dialogue systems with an increasing number of integrated features, implementation of adaptivity for turn-taking, and a firmer theoretical ground on which to build holistic dialogue architectures.

As many have argued, turn-taking is a fundamental and necessary mechanism for real-time verbal (and extraverbal) information exchange, and should, in our opinion, be one of the key focus areas for those interested in building complete artificial dialogue systems. Turn-taking skills include minimizing overlaps, minimizing silences, giving proper back-channel feedback, barge-in techniques, and other components which most people handle fluidly and with ease. People use various multimodal behaviors including intonation and gaze, for example, to signal that they have finished speaking and are expecting a reply (Goodwin, 1981). Based on continuously streaming information from our sensory organs, most of us pick up on such signals without problems, infer the correct state of dialogue, and what the other participants intend, and then automatically produce multimodal information in real-time that achieves the goals of the dialogue. In amicable conversations, participants usually share the goal of cooperation. Turn exchange—a negotiation-based activity based on the massive historical training ("socialization"') of the participants—usually proceeds so smoothly that people do not even realize the degree of complexity inherent in the processes responsible for making it happen.

The challenge of endowing synthetic agents with such skills lies not only in the integration of perception and action in sensible planning schemes but especially in the fact that these have to be tightly coordinated while marching to a real-world clock. How easy or difficult this is is dictated by the architectural framework in which the mechanisms are being implemented, and a prime reason for the broad overview we give of our dialogue architecture here.

In spite of recent progress in speech synthesis and recognition, lack of temporal responsiveness is one of a few key components that clearly sets current dialogue systems apart from humans; speech recognition systems that have been in development for over two decades are still far from addressing the needs of real-time dynamic dialogue (Jonsdottir et al., 2007). Many researchers have pointed out the lack of implemented systems intended to manage dynamic open-microphone/full-duplex dialogue (cf. Moore, 2007; Allen et al., 2001; Raux and Eskenazi, 2007), where the system is sufficiently aware of when it is given the turn, and can be naturally interrupted at any point in time by the human, and vice versa.

Although syntax, semantics, and pragmatics can indisputably play a large role in the dynamics of turn-taking, we have argued elsewhere that natural turn-taking is partially driven by a content-free planning system[2] (Thórisson, 2002b). People rely on signals and contextual cues that from the vantage point of humans are fairly primitive, e.g. prosody, speech loudness, gaze direction, facial expressions, etc. (Goodwin, 1981). In humans, recognition of prosodic patterns, based on the timing of speech loudness, silences, and intonation, is a more light-weight process than either word recognition, syntactic, or semantic processing (Card et al., 1986). Processing load between semantic processing and contextual/turn-signal processing is even more pronounced for artificial perception (the former being more computationally intensive than the latter), and therefore such cues represent prime candidates for inclusion in the process of recognizing turn signals in artificial dialogue systems. While in the future we intend to address the full scope of human turn management contextual cues, at present even these obvious ones present challenges to architectural and system design for real-time performance that must be overcome, and are therefore continuously addressed in our work.

In natural interactions, mid-sentence pauses are a frequent occurrence. Humans have little difficulty recognizing these from proper end-of-utterance silences,[3] and reliably determine the time at which it is appropriate to take turn—even on the phone, when no visual information is available. Temporal analysis of conversational behaviors in human discourse shows that turn-transitions in natural conversations most commonly take between 0 and 250 msecs (Stivers, 2009; Wilson and Wilson, 2005; Ford and Thompson, 1996; Goodwin, 1981) in face-to-face conversation. Silences in telephone conversations—when visual cues are absent—are at least 100 msecs longer on average (Bosch et al., 2005). In a study by Wilson and Wilson (2005), response time is measured in a face-to-face scenario where both parties always

[2] We use the term "planning" in the most general sense, referring to any system that makes a priori decisions about what should happen before they are put in action. By "content-free"' we mean, in short, virtually without consideration for the particular dialogue topic of a conversation.

[3] Silences are often not needed to signal end-of-turn in free-form human dialogue because the interlocutor derives it from other cues, such as prosody and content, often resulting in zero silence between turns (Goodwin, 1981).

had something to say. They found that 30% of between-speaker silences (turn-transitions) were shorter than 200 msecs and 70% shorter than 500 msecs. Within-turn silences, that is, silences where the same person speaks before and after the silence, are on average around 200 msecs but can be as long as 1 second, which has been reported to be the average "silence tolerance" for American-English speakers (Jefferson, 1989); longer silences are thus likely to be interpreted by a listener as a "turn-giving signal".[4] Tolerance for silences in dialogue varies greatly between individuals, ethnic groups, and situations; participants in a political debate exhibit a considerably shorter silence tolerance than people in casual conversation—this can further be impacted by social norms (e.g. relationship of the conversants), information inferable from the interaction (type of conversation, semantics, etc.), and internal information (e.g. mood, sense of urgency, etc.). To be on par with humans in turn-taking efficiency, a system thus needs to be able to predict, given an observed silience, what the interlocutor intends to do.

The motivation for the present work is to develop a complete conversational agent that can learn to interact and adapt its interaction behavior to its conversational partners, in a short amount of time. The agent may not know a lot about any particular topic of discussion, but it would be an "expert dialoguer", whose topic knowledge could be expanded as needed for various applications and as permitted by the artificial intelligence techniques under the hood. The Ymir Turn-Taking Model (YTTM) dialogue model (Thórisson, 2002b) proposes a framework for separating envelope control from topic control, making such an approach tractable. As a first step in this endeavour we are targeting a cooperative agent that can take turns, ideally with no speech overlap, yet achieves the shortest possible silence duration between speaker turns. Our approach is intended to achieve four key goals. *First*, we want to use on-line open-mic and natural speech when communicating with the system, integrating continuous acoustic perceptions as basis for decision making. We do not want to assume that the human must change their speech style or approach the system any differently than they do another human they might talk to. *Second*, we want to model turn-taking at a higher level of detail than previous attempts have done by including incremental perception and generation in a unified way. *Third*, because of the high individual variability in interaction style and pace, we want to incorporate *learning* from the outset, allowing the system to adapt to every person it interacts with

[4] "Turn-giving signals" are in quotes because they are not true "signals" in the engineering sense of the term, but rather socially conditioned "contexts"—combinations of features which together constitute "polite", "improper", "rude", or otherwise connotated contexts for the behavior of the interlocutors' behaviors.

on the fly. *Fourth*, we have argued elsewhere (Thórisson, 2008) that conversational skills—and by extension cognitive skills—allow for a high interconnectivity between its many functions; that they are a large, heterogeneous, densely coupled system (HeLD). The design of such HeLDs requires new architectural principles—standard software development methods will simply not suffice as they result in rigid systems and require more manpower for longer extended periods than any typical university or research lab is capable of securing. As a result, both the underlying software and conceptual architecture[5] must be highly modular, expandable and malleable. This approach puts a greater emphasis on methodology than is typical, but we believe it to be one of the few ways of actually achieving the integration of the many mechanisms necessary for creating a system approaching the flexibility and generality of real-world real-time human dialogue. It may also be considered of a "practical" nature, as it makes continuous expansion of the architecture more tractable for a small team. We have found architectural structure and makeup to greatly influence not only what kinds of operations it supports but also the speed of development and manageability. We see architectural design as a *necessary* part of any effort to develop dialogue systems intended to (incrementally) approach human dialogue skills.

The architecture described below thus rests on three main theoretical pillars. The first is a distributed-systems perspective,[6] the second relates to architectural software methodology, and the third is an underlying theory of turn-taking in multimodal real-time dialogue, outlined in Thórisson (2002b), emphasizing real-time negotiation as a key principle in turn-taking. In our approach, turn-taking negotiation is managed by time-dependent "cognitive contexts" (also called "fluid states" and "schema") that, for each participant, hold which perceptions and decisions are relevant or appropriate at each particular point in time, and represent the disposition of the system at any point in the dialogue, e.g. whether we might expect the other to produce a certain turn-taking cue, whether it is relevant to generate a particular behavior (e.g. volume increase in the voice upon interruption by the other, etc.).

[5] By "architecture" we mean the structure and operation of the system as a whole, containing many identifiable interacting parts whose organization essentially dictates how the system acts as a whole. The difference between software architecture and conceptual architecture is often subtle, but essentially is a separation between the operation of the particular software on the particular hardware and the behavior of the dialogue system it implements.

[6] By "distributed" we mean a system with multiple semi-independent processes that can be run on multiple CPUs, computers, and/or clusters.

Our current version of the system learns to become better at taking cooperative turns in real-time dialogue while it is up and running, improving its own ability to take turns *correctly* and *quickly*, with minimal speech overlap. The results are in line with prior versions of the system, where the system interacted with itself over hundreds of trials (Jonsdottir et al., 2008). Evaluation including human subjects so far includes a within-subjects study of 5 minutes of continuous interaction with each user (a total of 50 minutes), in three different conditions: (1) A closed, noise-free, setup with a very consistent interlocutor—another instance of itself ("Artificial" condition). (2) An open-mic setup, using Skype, where the system repeatedly interviews a fairly consistent interlocutor—the same human ("Single person" condition). (3) An open-mic setup, using Skype, with individual inconsistencies where the agent interviews 10 different human participants consecutively ("10 people" condition). The system adapts quickly and effectively (linearly) within 2 minutes of interaction, a result which, in light of most other machine-learning work on the subject—many of which require thousands of hand-picked training examples—is exceptionally efficient.

The rest of this chapter is organized as follows: First, we review related work, then we detail the architecture and learning mechanisms. A description of the evaluation setup comes next, followed by the results, summary, and future work.

2. Related Work

Models of dialogue produced by a standard divide-and-conquer approach can only address a subset of a system's behaviors, and are even quite possibly doomed at the outset. This view has been presented in our prior work (Thórisson, 2008) and is echoed in other work on dialogue architectures (cf. Moore, 2007). Requiring a holistic approach to a complex system such as human real-time dialogue may seem to be impossibly difficult. In our experience, and perhaps somewhat counterintuitively, when taking a breath-first approach to the creation of an architecture that models any complex system—where most of the significant high-level features of the system to be addressed are taken into account—the set of likely contributing underlying mechanisms will be greatly reduced (Schwabacher and Gelsey, 1996), quite possibly to a small, manageable set, thus greatly simplifying the task. It is the use of levels of abstraction that is especially important for cognitive phenomena: Use of hierarchical approaches is common in other scientific fields such as physics; for example, behind models of optics lie more detailed models of electromagnetic waves (Schaffner, 2006). A way to address the problem of building more complete models of dialogue is thus to take an interdisciplinary approach, bringing results from a number of sources to the table at various levels of abstraction and detail. This is essentially our approach.

When dealing with the modeling of complex phenomena, building architectures for systems that integrate multimodal data and exhibit heterogeneous real-time behaviors, it seems sensible to try to constrain the possible design space from the outset. One powerful way to do this is to build multilevel representations (cf. Schwabacher and Gelsey, 1996; Gaud et al., 2007; Dayan, 2000; Arbib, 1987); this may, in fact, be the only way to get our models right when trying to understand complex systems such as natural human dialogue. The thrust of this argument is not that multiple levels are "valid" or even "important", as that is a commonly accepted view in science and philosophy, but, rather, that to map correctly to the many ways sub-systems interact in such systems they are a *critical necessity*—that, unless our simulations are built at fairly high levels of fidelity, we cannot expect manipulations (expansions, modifications) by its designers to the architecture at various levels of detail to produce valid results. Modularity in the architecture is thus highly desirable as it brings transparency and openness to the architecture, making the modelling of a highly complex system tractable. However, gross modularity does not allow the kind of fine-grain representation that we argue is important for such systems. One drawback of fine-grain modularity is that decoupling components results in essence in a more distributed architecture, which calls for non-centralized control schemes. The kind of modularity and methodology one adopts is critical to the success of such decoupling.

Many of the existing methodologies that have been offered in the area of distributed agent-based system construction (cf. Wood and Deloach, 2000; Wooldridge et al., 2000) suffer from lack of actual use-case experiences, especially for artificial intelligence projects that involve construction of single-mind systems. We have built our present model using the Constructionist Design Methodology (CDM) (Thórisson et al., 2004) which helps us create complex multi-component systems at a fairly high level of fidelity, without losing control of the development process. CDM proposes nine iterative principles to help with the creation of such systems and has already been applied in the construction of several systems, both for robots and virtual agents (cf. Thórisson et al., 2004; Ng-Thow-Hing et al., 2007; Thórisson and Jonsdottir, 2008). CDM assumes a relatively manual construction process whereby a large number of pieces are integrated, for example speech recognition, animation, planning, etc., some of which may be off-the-shelf while others are custom-built. As such systems have to be deconstructed and reconstructed often, CDM proposes blackboards as the backbone for such integration. This makes it relatively easy to change information flow, add or remove computational functionality, etc., even at runtime, as we have regularly done.

As far as dialogue management and turn-taking are concerned, modular or distributed approaches are scarce. Among the few is the YTTM (Thórisson, 2002b), a model of multimodal real-time turn-taking. YTTM proposes that processing related to turn-taking can be separated, in a particular manner, from the processing of content (i.e. topic). Echoing the CDM, its architectural approach is distributed and modular and supports full-duplex multi-layered input analysis and output generation with natural response times (real-time). One of the background assumptions behind the approach, which has been reinforced over time by systems built using the approach (Thórisson et al., 2008; Jonsdottir, 2008; Ng-Thow-Hing et al., 2007), is that real-time performance calls for the incremental processing of interpretation and output generation.

The J.Jr. system (Thórisson, 1993) was a real-time communicative agent that could take turns in real-time casual conversation with a human. It was controlled by a finite state-machine architecture, similar to the Subsumption Architecture (Brooks, 1986). The system did not process the *content* of a user's speech, but instead relied on an analysis of prosodic information to make decisions about when to ask questions (i.e. take turn) and when to interject back-channel feedback. While modular, this architecture turned out to be difficult to expand into a larger, more intelligent architecture (Thórisson, 1996), especially when confronted with features at different time scales and levels of abstraction and detail (prosodic, semantic, pragmatic). Subsequent work on Gandalf (Thórisson, 1996) incorporated mechanisms from J.Jr. into the Ymir architecture, but presented a much more expandable, modular system of perception modules, deciders, and action modules in a holistic architecture that addressed content (interpretation and generation of meaning) as well as envelope phenomena (process control). A descendant of this architecture and methodology was recently used in building an advanced dialogue and planning system for the Honda ASIMO robot (Ng-Thow-Hing et al., 2007).

Raux and Eskenazi (2008) presented data from a corpus analysis of an online bus scheduling/information system, showing that a number of dialogue features, including speech act type, can be used to improve the identification of speech endpoint, given a silence. The authors tested their findings in a real-time system: Using information about dialogue structure—speech act classes, a measure of semantic completeness, and probability distribution of how long utterances go (but not prosody)—the system improved turn-taking latency by as much as 50% in some cases, but significantly less in others. This work reported no benefits from prosody for this purpose, which is surprising given that many studies have shown the opposite to be true (cf. Gratch et al., 2006; Schlangen, 2006; Thórisson, 1996; Traum and Heeman, 1996; Pierrehumbert and Hirschberg, 1990; Goodwin, 1981). We suspect one reason could be that the pitch and intensity extraction methods they used did not work very well on the data selected for analysis. Prosodic information has successfully been used to determine back-channel feedback in real-time. The Rapport Agent (Gratch et al., 2006) uses gaze, posture, and prosodic perception, among other things, to detect backchannel opportunities. The Ymir/Gandalf system (Thórisson, 1996) also used prosody, adding analysis of semantic, syntactic (and to a small extent even pragmatic) completeness to determine turntaking behaviors. Unfortunately evaluations of its benefit, for the purpose of turn-taking per se, are not available. The major lesson that can be learned from Raux and Eskenazi, echoing the work on Gandalf, is that turn-taking can be improved through an integrated, coordinated use of various features *in context*.

The problem of utterance segmenting for the purpose of proper turn-taking has been addressed to some extent in prior work. Of all the data sources informing dialogue participants about the state of the dialogue, prosody is the most prominent among the non-semantic ones. From the prior work reviewed, this seems like the most obvious place to start when attempting to design turn-taking mechanisms. Sato et al. (2002) use a decision tree to classify when silence signals that a turn should be taken. They annotated various features in a large corpus of human-human conversation to train and test the tree. The results show that semantic and syntactic categories, as well as understanding, are the most important features. These experiments have so far been limited to annotated data of a single, task-oriented domain. Applying their methods to a casual real-time conversation using today's speech recognition methods would inevitably increase the recognition time beyond any practical use because of an increased vocabulary—the content interpretation results could simply not be produced fast and reliably enough for making turn-taking decisions at sub-second speeds (Jonsdottir et al., 2007).

The introduction of learning into a dialogue system gives its designers yet another complex dimension which can affect everything and anything in the architecture's organization. Schlangen (2006) has successfully used machine learning to categorize prosodic features from corpus, showing that acoustic features can be learnt. Traum and Heeman (1996) have addressed the problem of utterance segmenting, showing that prosodic features such as boundary tones do play a role in turn-taking. As far as we know, none of this work has been applied to real-time situations. Bonaiuto and Thórisson (2008) demonstrate a system of two simulated interacting dialogue participants that learn to exploit each other's multimodal behaviors (that is, modality-independent multi-dimensional behaviors) to achieve a cooperative interaction where minimizing speech overlaps and speech pauses are the shared goals (as is the standard situation in amicable interactions between acquaintances, friends, and family—shared with the present work). Using a neuro-cognitive model of learning, the work shows that emergent properties of dialogue, pauses, hesitations, interruptions—i.e. negotiations of turn—can be learned via the general framework provided by YTTM, and its fluid states, coupled with Bonaiuto and Arbib's ACQ model of learning (Bonaiuto and Arbib, 2010). While Bonaiuto and Thórisson's system was based on the YTTM, the implementation of the learning mechanisms was neither meant to run on-line nor in real-time.

In summary, no prior system has implemented a comprehensive dialogue system capable of on-line learning of turn-taking skills, and allowed it to adapt to its interlocutors in real-time. The turn-taking model presented here is an extended version of the YTTM (Thórisson, 2002b) with the simplification that the communicative channel is limited to the speech modality. Turn-taking is modeled as an agent-oriented negotiation process with eight turn-taking, semi-global "cognitive contexts" or fluid states that define the perceptual and behavioral disposition of the system at any point in the dialogue, as already mentioned. These contexts support, in effect, a distributed planning and control system for both perception and action; the distributed learning scheme we present below implements a negotiation-driven tuning of real-time turn-taking behaviors within this framework.

3. System Architecture

Our multi-module dialogue system is capable of real-time dialogue with human users speaking naturally, with no artificial constraints on the process of interaction. As mentioned above, the architecture follows the principles of modularity outlined above, as specified by the CDM methodology (Thórisson et al., 2004; Thórisson, 2008), and enables us to introduce learning into the architecture in a modular

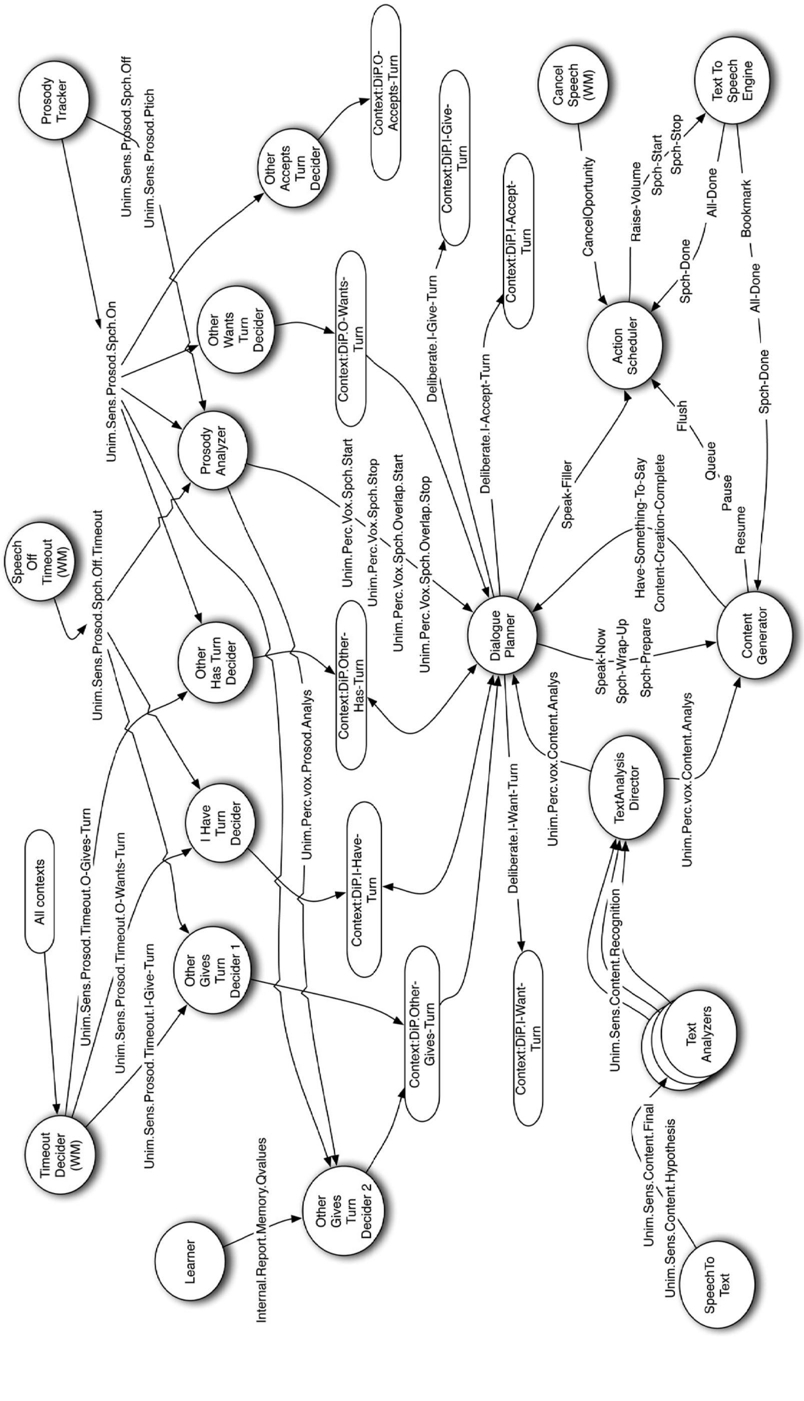

Figure 1. Flat layout of message passing between modules.

way.[7] As an indication of the architecture's present scope, Figure 1 presents a full (flat) view of the system's gross architecture. (Note that while a flat view is informative of the architecture's scope, it belies its naturally hierarchical nature—an important feature of our system that allows us to build a complex architecture with a very small team of developers.) We will discuss the architecture's various components below. A complete introduction to the architecture is beyond the scope of this chapter; the main focus of this chapter will be on the parts of turn-taking needed to support learning of efficient turn-taking.

Following the Ymir architecture (Thórisson, 1996), our system's modules are categorized based on their functionality; perception-, decider-, and action modules, at the coarsest granularity (see Figure 2). We will now describe the modules of these types that relate to the turn-taking system.

3.1 Perception

As already mentioned, although the architecture is inherently a multimodal system (as shown in prior work (c.f. Bonaiuto and Thórisson, 2008; Ng-Thow-Hing et al., 2007; Thórisson, 2002b, 1996)), the current system's input is limited to audio input. There are two main perception modules that deal with prosodic features in the system, the Prosody Tracker and the Prosody Analyzer. The Prosody Tracker is a low-level perception module whose input is a raw audio signal (Nivel and Thorisson, 2008). It computes speech signal levels and determines information about speech activity, producing time-stamped Speech-On and Speech-Off messages. It also analyzes the speech pitch incrementally (in steps of 16 msecs) and produces pitch values, in the form of a continuous stream of pitch message updates.

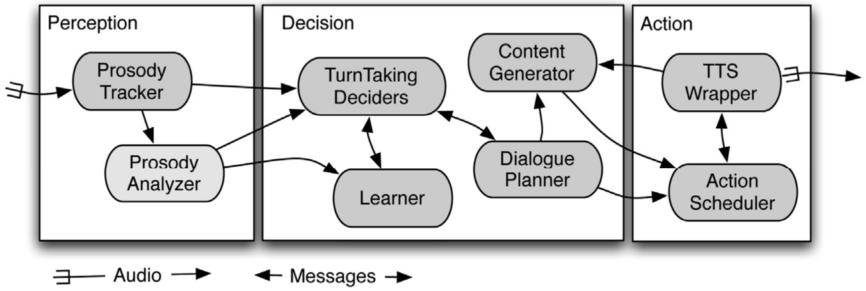

Figure 2. System components, each component consists of one or more modules.

[7] This is important because it not only allows the architecture to be more easily expanded in the future, but also the learning mechanisms, which we will show in future papers.

Similar to Thórisson (2002a), pitch is further analyzed by a Prosody Analyzer perception module to compute a more compact representation of the pitch pattern in a discrete state space, in our case to support the learning: The most recent tail of speech right before a silence, the last 300 msecs, is analyzed to detect minimum and maximum values of the fundamental pitch to produce a tail-slope pattern of the pitch. Slope is split into semantic categories; in the present implementation we have used three categories for slope: *Up, Straight* and *Down* according to Formula 1 and three for the relative value of pitch right before silence: *Above, At* and *Below*, as compared to the average pitch according to Formula 2.

$$m = \frac{\Delta pitch}{\Delta msecs} \begin{cases} if\ m > 0.05 \rightarrow slope = Up \\ if\ (-0.05 \leq m \leq 0.05) \rightarrow slope = Straight \\ if\ m < 0.05 \rightarrow slope = Down \end{cases} \tag{1}$$

$$d = pitch_{end} - pitch_{avg} \begin{cases} if\ d > Pt \rightarrow end = Above \\ if\ (-Pt \leq d \leq Pt) \rightarrow end = At \\ if\ d < Pt \rightarrow end = Below \end{cases} \tag{2}$$

where *Pt* is the average ± 10, i.e. pitch average with a bit of tolerance for deviation.

The primary output of the Prosody Analyzer is a symbolic representation of the particular prosody pattern identified in this tail period (see Figure 3). More features could be added into the symbolic representation, with the obvious side effect of increasing the state space.

The Speech-To-Text module and Text Analyzers deal with speech recognition. Speech recognition is done incrementally with the best

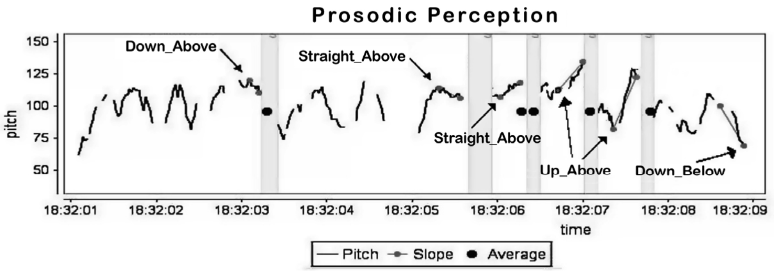

Figure 3. A window of 9 seconds of spontaneous speech, which includes speech periods and silences, categorized into descriptive groups for slope and end position relative to the average pitch. Only slope of the fundamental pitch during the immediate 300 msecs preceding a silence (indicated by the gray area) is categorized (into Up, Straight, and Down). (Abscissa: Voice F0 in Hz, as produced in near real-time by Prosodica; mantissa: Time-Hours/minutes/seconds.)

score hypothesis being available to the rest of the system during interlocutors' speech, but final utterance is not calculated until at least one second of silence has been detected.

3.2 Deciders

Our detailed turn-taking model consists of eight dialogue states (see Figure 4). This represents the states taken when the turn switches hands. The dialogue states are modeled with a distributed semi-global context system, implementing what can (approximately) be described as a distributed finite state machine that selectively applies to the activation and de-activation of most modules in the system. Context transition control ("state transitions") in this system is managed by a set of deciders (Thórisson, 2008). There is no theoretical limit to how many deciders can be active for a single given system-wide context. Likewise, there is no limit to how many deciders can manage identical or non-identical transitions. Reactive deciders (IGTD, OWTD, ...) are the simplest, with one decider per transition. Each contains at least one rule about when to transition, based on both temporal and other information. Transitions are made in a pull manner: the Other-Accepts-Turn-Decider, e.g. transits to context Other-Accepts-Turn (see Figure 4).

The Dialogue Planner (DP) and Learning modules (see further description below) can influence the dialogue state directly by sending context transition messages I-Want-Turn, I-Accept-Turn, and I-Give-Turn; however, all these decisions are under the supervisory control of the DP: If the Content Generator (CG) has some content ready to be communicated, the agent might want to signal that it wants a turn and it may want to signal I-Give-Turn when the content queue is empty (i.e. have nothing to say). Decisions made by these modules override decisions made by other turn- taking modules. The DP also manages the content delivery; that is, when to start speaking, withdraw, or raise one's voice. The CG is responsible for creating utterances incrementally, in "thought chunks", typically of durations shorter than 1 second. We are developing a dynamic content generation system at present; based on these principles the CG currently simulates its activity by selecting thought units to speak from a pre-defined list. It signals when content is available to be communicated and when content has been delivered.

In the present system, the module Other-Gives-Turn-Decider-2 (OGTD-2) uses the data produced by the Learner module to change the behavior of the system. At the point when the speaker stops speaking, the challenge for the listening agent is to decide how long to wait before starting to speak (OGTD-1 has a static behavior of transitioning

Reactive deciders	
I-Have-Turn-Decider (IHTD)	Other-Has-Turn-Decider (OHTD)
I-Accept-Turn-Decider (IATD)	Other-Accepts-Turn-Decider (OATD)
I-Give-Turn-Decider (IGTD)	Other-Gives-Turn-Decider (OGTD)
I-Want-Turn-Decider (IWTD)	Other-Wants-Turn-Decider (OWTD)

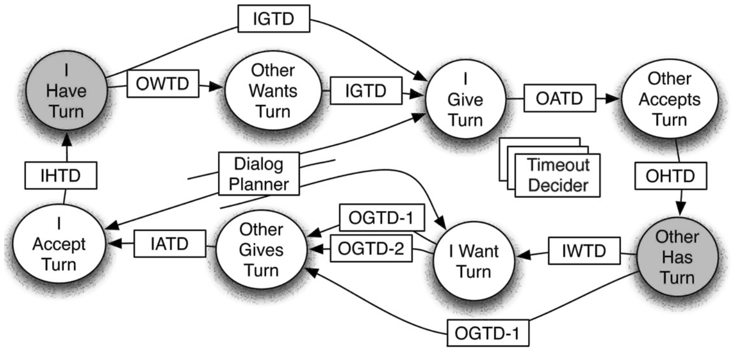

Figure 4. The heart of turn-taking control in the system consists of a set of eight semi-global context-states and 11 deciders. In context-state I-Have-Turn (IHT), both I-Give-Turn-Decider (IGTD) and Other-Wants-Turn-Decider (OWTD) are active. Unlike other modules, the Dialog Planner (DP) can transition independently from the system's current context-state and override the decisions from the reactive deciders. A Timeout-Decider handles transitions if one of the negotiating contexts is being held unacceptably long (but its transitions are not included in this diagram; also not shown are which modules are active during which contexts).

to Other-Gives-Turn after a two-second silence). If the agent waits too long, and the speaker does not continue, there will be an unwanted silence; if he starts too soon and the speaker continues speaking, overlapping speech will result. We solve this by having OGTD-2 use information about past prosody, which occurs right before the latest silence, to select an optimal silence tolerance window (STW), as will now be described in detail.

4. The Learner

The learning mechanism is implemented as an independent component (Learner module) in the modular architecture described above. It is based on the Actor-Critic distribution of functionality (Sutton and Barto, 1998), where one or more actors make decisions about which actions to perform and a critic evaluates the effect of these actions on the environment; the separation between decision and action is important because in our system a decision can be made to act in the future. In the highly general and distributed learning mechanism we have implemented, any module in the system can take the role of an actor by sending out decisions and receiving, in return, an updated decision policy from an associated Learner module. A decision consists of a state-action pair: the action being selected and the evidence used in making that action represents the state. Each actor follows its own action-selection policy, which controls how it explores its actions; various methods such as å-greedy exploration, guided exploration, or confidence value thresholds can be used (Sutton and Barto, 1998).

In our system, the Learner module takes the role of a critic. It consists of the learning method, reward functions, and the decision policy being learnt. A Learner monitors decisions being made in the system and calculates rewards based on a reward function, a list of decision/event pairs, and signals from the environment—in our case overlapping speech and long silences—and publishes an updated decision policy (the environment consists of the relevant modules in the system), which any actor module can subsequently use to base its decision on.

We use a delayed one-step Q-Learning method according to the formula:

$$\mathbf{Q(s, a) = Q(s, a) + \alpha[reward - Q(s, a)]} \qquad 3$$

where *Q*(*s*,*a*) is the learnt estimated return for picking action *a* in state *s*, and *?* is the learning rate. The reward functions—what events following what actions lead to what reward—is pre-determined in the Learner's configuration in the form of rules: A *reward* of *x* if *event y* succeeds at *action z*. Each decision has a lifetime in which system events can determine a reward, but the reward can also be calculated in absence of an event, after its given lifetime has passed (e.g. no overlapping speech). Each time an action gets a reward, the return value is recalculated according to Formula 3 and the Learner broadcasts the new value.

In the current setup, Other-Gives-Turn-Decider-2 (OGTD-2) is an actor in Sutton's sense (Sutton and Barto, 1998); it decides essentially what its name implies. This decider is only active in the state I-Want-Turn. It learns an "optimal" STW, which prevents it from speaking on top of the other, while minimizing the lag in starting to speak, given a silence. Each time a Speech-Off signal is detected, OGTD-2 receives analysis of the pitch in the last part of the utterance preceding the silence from the Prosody Analyzer. The prosody information is then used to represent the state for the decision; a predicted safe STW is selected as the *action* and the Decision is posted. The end of the STW determines when, in the future, the participant who currently doesn't have the turn will start speaking (take the turn). In the case where the interlocutor starts speaking again before this STW closes, the decider doesn't signal Other-Giving-Turn, essentially canceling the plan to start speaking (see Figure 5). This leads to a better reward, since no overlapping speech occurs. If he starts talking just after the STW closes, after the decider signals Other-Gives-Turn, overlapping speech will likely occur (keep in mind that, due to processing time, once a decision has been made it can take time before it is actually executed), leading to negative reinforcement for this size of STW given to the particular prosodic information observed.

This learning strategy is based on the assumption that both agents want to take turns cooperatively ("politely") and efficiently. We have already begun expanding the system to be able to interrupt dynamically and deliberately—i.e. be "rude"—and the ability to switch back to being polite at any time, without destroying the learned data. This research will be discussed at a later date.

5. Quantitative Evaluation of Learning

We will look at system performance across three dependent measures:

- The system's ability to select an appropriate STW. Given a silence in the user's speech, the selection of an STW is based on the type of prosody pattern perceived right before the silence. If turn-giving indicators are perceivable to the system, we should find clear variations in STW lengths based on the pattern perceived. If no evidence of turn-giving is detected by the system, we should find an even distribution of STW size between patterns.

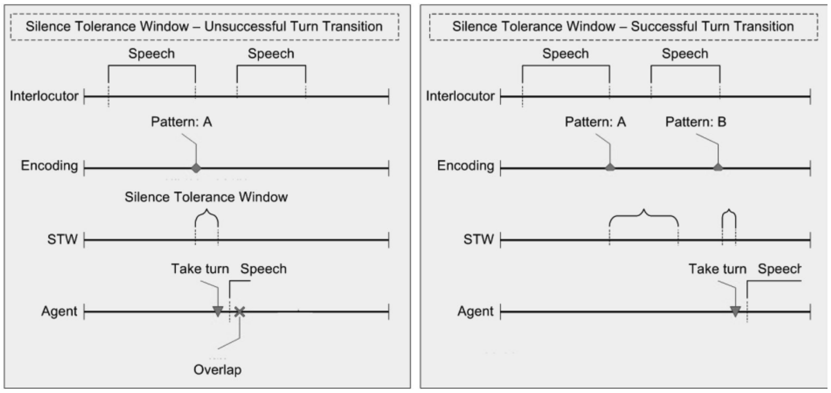

Figure 5. The interlocutor's speech is analyzed in real-time; as soon as a silence is detected the prosody preceding the silence is decoded. The system makes a prediction by selecting an STW, based on the prosody pattern perceived in the interlocutor. This window is a prediction of the shortest safe duration to wait before taking turn: A window that is too short will probably result in overlapping speech while a window that is too large may cause unnecessary or unwanted silences.

- How quickly the agent takes its turn. We evaluate this by measuring the length of the silence before each successful turn-transition (from other to the agent) and compare the results to human data.
- Frequency of overlapping speech. Because the agent should be learning to be polite—i.e. not speak on top of the other—the number of overlaps should decrease over time. (Note: In our Speaking-with-Self condition we use a closed sound loop (no open mic), but an open mic setup when the system speaks with humans.)

5.1 Hypotheses and statistics

To evaluate the learning mechanism, we used linear regression on the single-case data sessions (*Artificial*—talking to itself (a copy of itself in the interviewee role) for 10 consecutive sessions with 30 questions each; *Single person*—talking to one person for 10 consecutive sessions with 30 questions each). For the *10-person* condition (asking 10 different people 30 questions each), we used within-subject *t*-tests between the first five sessions and the second five sessions. In all cases the dependent variables are: (a) Taking Turn in less than 500 msecs, (b) Taking Turn in less than 300 msecs, and (c) Number of Overlaps.

The hypotheses are:

- H1: Frequency of taking turn within less than 500 msecs should increase as a function of number of turns.
- H2: Frequency of taking turn within less than 300 msecs should increase as a function of number of turns.
- H3: Frequency of overlapping speech should be higher in the first half of the interviews than in the second half of the interviews.

5.2 Interview setup

The agent is configured to ask 30 pre-defined questions, using, among other things, STW to control its turn-taking behavior during the interlocutor's turn (see Figure 5). Each interaction takes approximately five minutes. We have run three different evaluation conditions with the system.

1. **The system interviewing itself ("Artificial").** Having a single artificial interlocutor interacting with a non-learning instance of itself gives us a very consistent behavior in a setup with no background noise, providing a baseline for the real-world evaluations.

2. **The system interviewing a single person ("Single person").** A single person should be fairly consistent in behavior, but some external noise is inevitable since the communication is through Skype. Significant results with a single person would show that the system can adapt with a very small set of learning data—a highly desirable feature for such systems.
3. **The system interviewing 10 people ("10 people").** This is the most complex condition, as there is both individual variation between participants as well as background noise. Individual variations could be a confounding factor; getting significant results in this condition would mean that the system shows robustness to individual variation. Improvement over time indicates that the system can learn from, and in spite of, individual differences.

In all conditions, the system is learning to take turn in a "polite" cooperative manner while striving for the shortest possible silence between turns. Each evaluation consists of 10 consecutive interviews. Our learning system, named Askur for convenience, begins the first interview with no knowledge, and gradually adapts to its interlocutors throughout the 10 interview sessions.

The goal of the learning system is to learn to take turns with no speech overlap, yet achieve the shortest possible duration of silence between speaker turns. To eliminate variations in STW due to lack of something to say, we have chosen an interview scenario where the learning agent is the interviewer, in which case it always has something to say (until it runs out of questions and the interview is over).

We are aiming at an agent that can adapt its turn-taking behavior to dialogue in a short amount of time, using incremental perception. In the evaluations we focus exclusively on detecting turn-giving indicators in deliberately generated prosody, leaving out the topic of turn-opportunity detection (i.e. turn transition without prior indication from the speaker that she's giving the turn), which would, for example, be necessary for producing human-like interruptions.

A sample of 11 Icelandic volunteers took part in the experiment, none of whom had interacted with the system before. All subjects spoke English to the agent, with varying amounts of Icelandic prosody patterns, which differ from native English-speaking subjects. The study was done in a partially controlled setup; all subjects interacted with the system through Skype using the same hardware (computer, microphone, etc.) but the location was only semi-private and some background noise was present in all cases.

5.3 Parameter settings

The main goal of the learning task is to differentiate silences in real-time based on partial information of an interlocutor's behavior (prosody only) and predict the best reciprocal behavior. For best performance, the system needs to find the right tradeoff between shorter silences and the risk of overlapping speech. To formulate this as a Reinforcement Learning problem, we need to define states and actions for our scenario.

Using single-step Q-Learning, the feature combination in the prosody preceding the current silence becomes the *state* and the length of the STW becomes the *action* to be learned. For efficiency, we have split the continuous action space into discrete logarithmic values (see Table 1), starting with 10 msecs and doubling the value up to 1.28 seconds (the maximum STW where the system takes the turn by default). The action selection policy for OGTD-2 is ε-greedy with 10% exploration, always selecting the shorter STW if two or more actions share the top spot.

The reward given for decisions that do not lead to overlapping speech (i.e. successful transitions) is the milliseconds in the selected STW; a 100 msec STW will receive a reward of –100 if successful and STW of 10 msecs will receive –10 points. If, however, overlapping speech results from the decision (i.e. the action is unsuccessful), a fixed reward of –2000 (i.e. more than waiting the maximum amount of time) is given. This is to simulate that when two STWs are without overlap, the smaller is better. Every reward in the learning system is negative, resulting in unexplored actions being the best option at each time, since return starts at 0.0 for unexplored actions, and once a reward has been given the return can only decrease. In the beginning, the agent

Table 1. Discrete actions representing STW size in msecs.

Action (STW)	Reward: Successful transition	Reward: Unsuccessful transition
10	–10	–2000
20	–20	–2000
40	–40	–2000
80	–80	–2000
160	–160	–2000
320	–320	–2000
640	–640	–2000
1280	–1280	–2000

is only aware of actions 1280 and 640 and only explores shorter STWs for patterns where the lowest available STW is considered the best.

6. Results

To reiterate, there are three conditions: Artificial, Single person, and 10 people. First we will answer the question of whether the system is learning; then we will look at the above dependent measures in more detail.

6.1 Is the system learning?

The system showed significant learning effects for the Artificial condition, both for reaction time (simple regression F = 12.83; $p <$ 0.0005) and overlaps (simple regression F = 10.41; $p <$ 0.0047). The system also showed significant learning effects for the 10-person condition, for reaction time (see Table 2), and overlaps (see Table 3). Although an 89 msec gain in STW may seem small, it makes a big qualitative difference for most average dialogue participants, essentially changing an automatic dialogue system from being obviously inadequate and sometimes annoyingly slow to not being so. The system starts each interview with previous learning and thus optimal STW based on another person's prosody patterns instead of beginning with a "safe" 1-2 second STW. To shorten this previous optimal STW, at the same time as overlaps drop from 24% to 10%, shows that the agent is learning new skills on the fly, becoming increasingly more "polite" (efficient and cooperative) by improving its reaction time and speech overlap performance between- as well as within-interviews.

Table 2. Paired one-tail *t*-test: Interviewing 10 consecutive people.

Turn	Observation (*N*)	Mean	St.Dev
Turn 1–15	10	655 msecs	137.25
Turn 16–30	10	566 msecs	73.83
***T*-value = 2.46, *P*-value = 0.018, DF = 9**			

Table 3. Paired one-tail *t*-test: Overlaps when interviewing 10 consecutive people.

Turn	Observation (*N*)	Mean	St.Dev
Turn 1–15	10	0.24	0.11
Turn 16–30	10	0.10	0.09
***T*-value = 4.16, *P*-value = 0.0012, DF = 10**			

In the single-person condition, overlaps get continually fewer (simple regression $F = 3.39$; $p < 0.08$), but improvement in reaction time is not statistically significant although still indicative of the same trends as observed in the other conditions. The observed improvements are nevertheless in the expected direction, indicating that the system is, in fact, improving during its interactions with this particular individual, as it did with statistical significance for the more consistent artificial interlocutor. In future we will seek to improve the performance for individuals, since a noticeable adaptivity at the individual level is a worthy, quite impressive goal to reach.

6.2 Silence tolerance window (STW) by pattern

We look for turn-giving intonation patterns in the last 300 msecs of speech before each silence. Tail pattern of the pitch is currently categorized into nine semantic categories based on slope (Up, At, Down) and final pitch compared to average (Above, At, Below).

To begin with, we will analyze the distribution of these patterns before the silences that mark end of turn, and before the silences that are within turn. In both the artificial interviewee evaluation and single-person evaluation, the pattern *Down_Below* (representing a final fall in pitch) is most widely used at end of a turn (see Table 4). This harmonizes well with previous research (Pierrehumbert and Hirschberg, 1990), which has associated a final fall in pitch with a turn-giving signal. Furthermore, the person and the artificial interviewee have a very similar distribution of patterns at the end of a turn. The same cannot be said about prosody patterns perceived before silences that do not lead to turn transition. Prosody before silences within

Table 4. Distribution between prosody patterns.

Pattern	Artificial Interlocutor		Human Interlocutor	
	At end	Within	At end	Within
Down_Below	58.6%	0.4%	42.0%	12.6%
Straight_Below	10.3%	0.1%	14.1%	17.2%
Up_Below	8.6%	2.4%	7.3%	3.6%
Down_At	8.4%	20.6%	10.4%	14.6%
Up_Above	5.6%	38.1%	5.0%	14.4%
Straight_At	2.7%	10.6%	6.5%	15.7%
Straight_Above	2.2%	13.7%	7.3%	9.7%
Down_Above	2.1%	10.2%	3.4%	4.6%
Up_At	1.5%	4.1%	3.9%	7.6%

turn are much more evenly distributed between categories in the person's speech than in the artificial interviewee's speech. The artificial interviewee is as stated before very consistent, he decides what to say beforehand and sticks to that. After listening to the recordings of the person speaking there is a lot more variation occurring; decisions are being made and changed at the spur of the moment leading to more inconsistencies in prosody. An example of that is a person giving a short answer with prosody that can be perceived as giving turn and then adding to the answer and again ending with a give-turn prosody (e.g. "My favorite actor is Will Smith. and Ben Affleck.").

When the agent interviews 10 consecutive people, we analyzed which patterns were most widely used at the end of a turn. We found that four patterns out of nine are seen in up to 80% of turn-transitions (see Figure 6). None of these patterns have an end pitch above session average. This might be due to the fact that people are not asking the agent any questions—and questions tend to end on a higher-than-average pitch.

We further analyzed the use of the final fall pattern *Down_Below*, both as turn-transition and within turn. The use of final fall, both at end of turn and within turn, varies considerably between participants. The person that uses final fall the most at end of turn uses it in 41% of the ends of turns, while the person that uses it the least only uses it in 2.7% of cases (see Table 5). This is surprising as the pool of participants are all from the same cultural background and we would thus speculate more similarities in behavior.

Table 5. Usage of Down Below in the 10-person study.

Participant	At end	Within
1	7.7%	14.9%
2	14.8%	7.3%
3	34.8%	6.7%
4	6.3%	9.1%
5	2.7%	7.1%
6	27.3%	15.4%
7	41.0%	8.7%
8	18.8%	5.0%
9	11.1%	2.5%
10	25.0%	19.2%

6.3 Silence length

A study on human behavior by Wilson and Wilson (2005) measured silences in face-to-face conversations where participants always had something to say. They reported response time to be shorter than 500 msecs in 70% of turn-transitions and shorter than 200 msecs in 30% of turn-transitions.

Our study was conducted over a relatively low-quality voice connection (Skype) and not face-to-face, and thus allows only for voice cues to communicate envelope feedback regarding turns. The studies are compatible in the sense that our agent always has something to say, while people might have to think a bit before they answer. Silences in telephone conversations tend to average about 100 msecs longer than in face-to-face conversations (Bosch et al., 2005), so we have measured silences shorter than 300 msecs and shorter than 500 msecs.

Table 6. Average silences for each condition.

Condition	Shorter than 500 msecs	Shorter than 300 msecs
Artificial	53.1%	32.2%
Single person	44.0%	16.3%
10 people	43.7%	8.4%

Our agent takes its turn in less than 500 mescs in 53.1% to 43.7% of turns for our three conditions (see Table 6). This is the average for the last nine interviews, eliminating the first interview due to the preset STW of 1280 and 640 msecs, which would interfere with obtaining results based on real-time interactions.

When looking at how silence length evolves during the series of interviews, it is obvious that Askur (the interviewer) adapts relatively quickly in the beginning in all cases. In the first session, when the agent interviews a copy of itself in the interviewee role, it is obviously interviewing the most consistent interviewee; the agent gets constantly better with only minor lapses until it reaches about 70% of silences shorter than 500 msecs and around 40% of silences shorter than 300 msecs (see Figure 7). When interviewing a single person for 10 consecutive interviews, the system cannot learn as well as when interviewing a copy of itself since there is more variation in behavior.

When interviewing 10 people, Askur has reached about 50% of before-turn silences shorter than 500 msecs (see Figure 7), compared to 70% in the human-human comparison data. There are two distinct dips in performance in interviews four and eight. These can be attributed to differences in the prosody patterns used by participants (see Figure

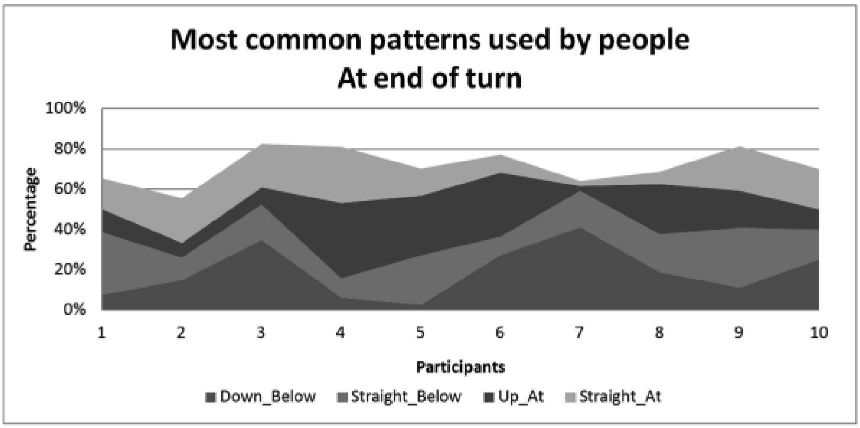

Figure 6. Four prosody categories out of nine are seen in up to 80% of turn-transitions before the agent takes turn.

6). In the case of interviewee four, the agent needs to learn that *Up_At* is a turn giving signal (used in 37.5% of 4's turns), but in the case of participant eight it is not as obvious. While examining overlaps, it can be seen that a lot of overlaps occurr in interview eight and at the beginning of interview nine, indicating that the agent is making mistakes (see Figure 8).

6.4 Turn overlaps

The final evaluation of success is to view the overlapped turns in each condition. In the first condition when interviewing self (Artificial), the overlaps are mostly in the first half of the evaluation. After that, overlaps drop considerably and stay low throughout the remainder of the sessions. This is due to the consistency of the interlocutor, the system learns how to interact with the interlocutor, and makes very few mistakes towards the end of the evaluation. In the second condition (Single person) when interviewing a single person for 10 sessions, overlaps are around 10% or below for all interviews except at the beginning of third and fifth interviews. In the last scenario where the system interviews 10 different people, overlaps occur more randomly due to differences in participants.

It is not surprising that most overlaps are perceived in the last condition, when the system interviews 10 different people (17%). It is, however, surprising that fewer overlaps are perceived when interviewing a single person over an open microphone than when interviewing an artificial interlocutor in a closed (sound card to sound card) setting (see Table 7). The artificial interlocutor always selects one to three sentence fragments and inserts artificial "think pauses"

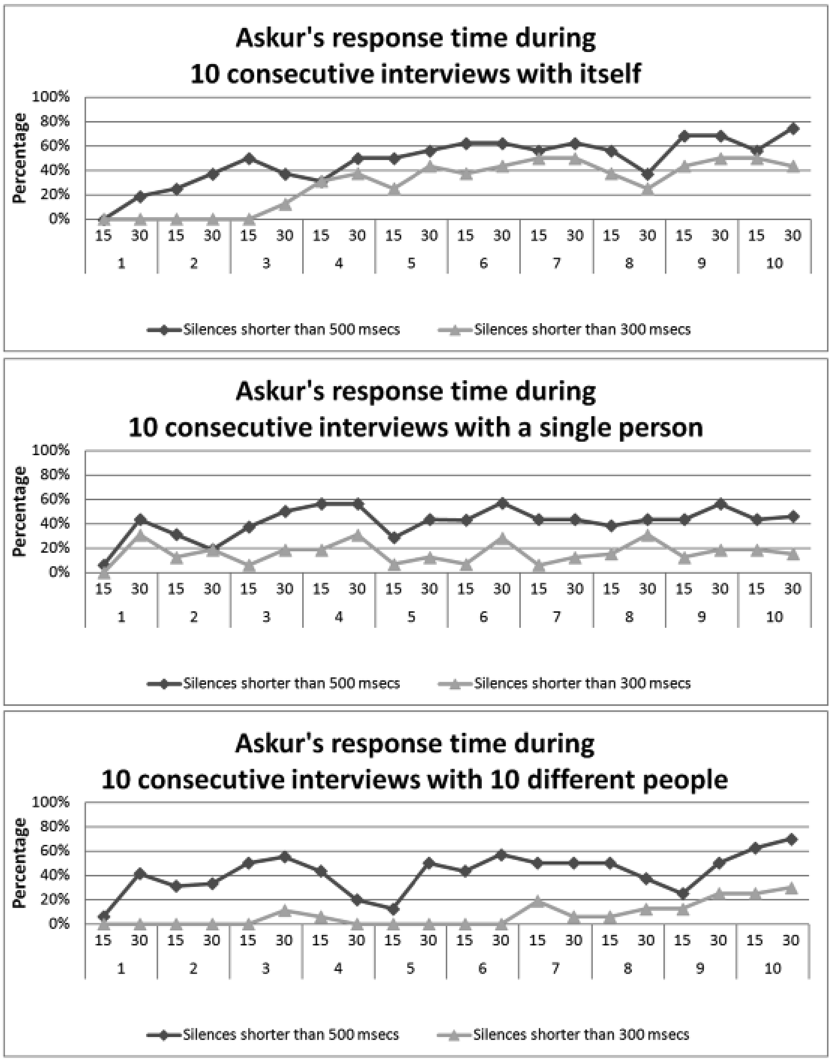

Figure 7. Proportion of silences with human speed characteristics. The graphs show 10 consecutive interviews in three different conditions. Each session is 10 consecutive interviews, each interview is 30 turns.

Table 7. Average silences for each condition.

Condition	Overlapped turns
Artificial	15.3%
Single person	10.3%
10 people	17.0%

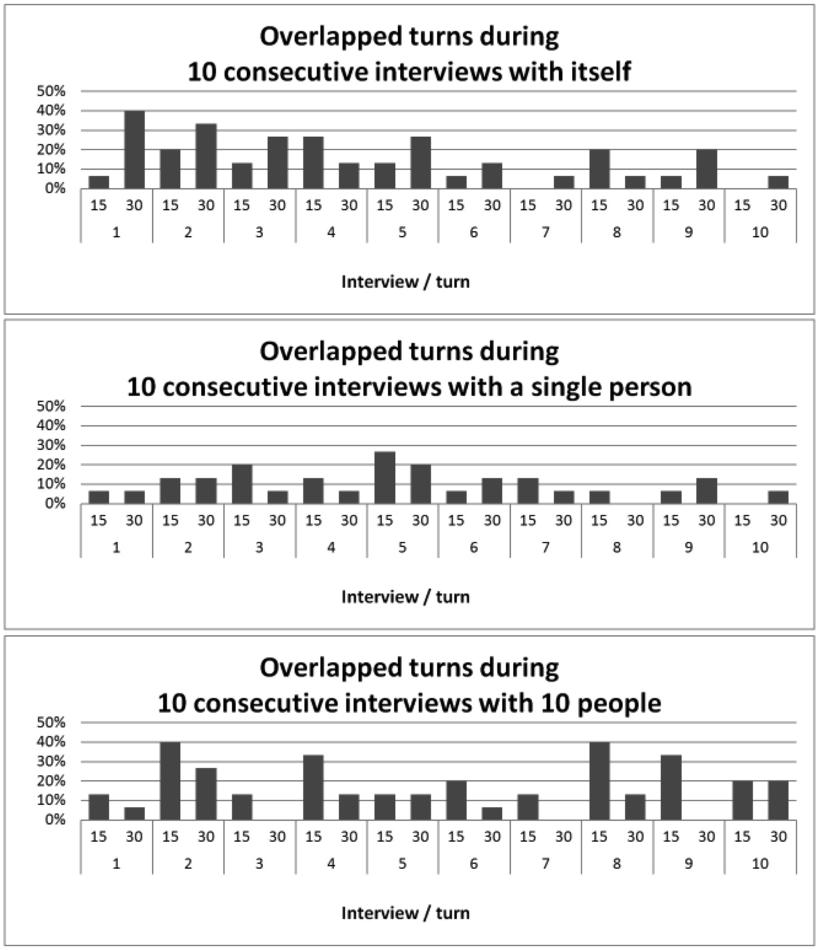

Figure 8. Overlapped turns in our three evaluations.

with a length 0 to 1000 msecs between them; people tend to answer in shorter sentences, not allowing for as many opportunities for mistakes.

7. Conclusions and Future Work

Our system learns to optimize STW and minimize speech overlaps and awkward silences, using prosody analysis to predict interlocutor behavior. It learns this on the fly, in a full-duplex "open-mic" (dynamic interaction) setup, and can take turns very efficiently in dialogues with copies of itself and with people, in relatively human-like ways.

The system finds prosodic features that can serve as predictors of human turn-giving behavior, and employs incremental (real-time) perception to work in as close to human natural dialogue speeds as possible. As the system learns on-line, it is able to adjust quite quickly to the particulars of individual speaking styles. At present, the system strongly targets the temporal characteristics of human-human dialogue, something that is mostly considered irrelevant by prior and related work on dialogue systems, as the above discussion shows. While the results are encouraging, there is room for significantly more work to be done in this direction.

At present, the system is limited in two main ways: it assumes a small set of turn-taking circumstances where content does not play a role and a single shared goal of cooperative "polite" conversation is assumed, where both parties want to minimize speech overlaps. Silences caused by outside interruptions—e.g. barge-in techniques and deliberate interruption techniques—are therefore a topic for future study. The system is highly expandable, however, as it was built as part of a much larger system architecture that addresses multiple topic- and task-oriented dialogue, as well as multiple communication modes such as gesture and facial expression. In the near future, we expect to expand the system to more advanced interaction types and situations. The learning mechanism described here will be expanded to learn not just the shortest durations but also the most efficient turn-taking techniques in multimodal interactions under many different conditions.

Because of the distributed nature of the architecture, the turn-taking system is constructed in such a way as to allow a mixed-control relationship with outside processes. This means that we can expand it to handle situations where the goals of the dialogue may be very different from being "friendly", even adversarial, as, for example, in on-air open-mic political debates. How easy this is remains to be seen; the main question revolves around the learning systems—how to manage learning in multiple circumstances without negatively affecting prior training.

Acknowledgement

This work was supported in part by research grants from RANNIS, Iceland, and by a Marie Curie European Reintegration Grant within the 6th European Community Framework Programme. The authors wish to thank Yngvi Björnsson for his contributions to the development of the reinforcement mechanisms.

REFERENCES

Allen, J.F., G. Ferguson and A. Stent. 2001. An architecture for more realistic conversational systems. Proceedings of the 6th international conference on Intelligent user interfaces, pp. 1–8.

Arbib, M. 1987. Levels of modeling of visually guided behavior. *Behavioral and Brain Sciences*, **10(3)**:407–465.

Bonaiuto, J. and K.R. Thórisson. 2008. Towards a neurocognitive model of realtime turn taking in face-to-face dialogue. In Wachsmuth, I., Lenzen, M. and Knoblich, G. (eds.), Embodied Communication in Humans and Machines. UK: Oxford University Press, pp. 451–485.

Bonaiuto, J. and M. Arbib. 2010. Extending the mirror neuron system model, II: what did I just do? A new role for mirror neurons. *Biological Cybernetics*, **102(4):**341–359.

Bosch, L.T., N. Oostdijk and L. Boves. 2005. On temporal aspects of turn taking in conversational dialogues. *Speech Communication*, **47(11-2):**80–86.

Brooks, R.A. 1986. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, **2(1):**14–23.

Card, S.K., T.P. Moran and A. Newell. 1986. The model human processor: An engineering model of human performance. In K.R. Boff, L. Kaufman and J. P. Thomas (eds.), Handbook of Perception and Human Performance. Vol. 2: Cognitive Processes and Performance, 1986, pp. 1–35. Handbook of Human Perception, volume II. New York: John Wiley and Sons.

Dayan, P. 2000. Levels of analysis in neural modeling. Encyclopedia of Cognitive Science. London, England: MacMillan Press.

Ford, C. and S.A. Thompson. 1996. Interactional units in conversation: Syntactic, intonational, and pragmatic resources for the management of turns. In Ochs, E., Schegloff, E. and Thompson, S.A. (eds.), Interaction and Grammar, pp. 134–184. Cambridge: Cambridge University Press.

Gaud, N., F. Gechter, S. Galland and A. Koukam. 2007. Holonic multiagent multilevel simulation: Application to real-time pedestrian simulation in urban environment. *Procedings of International Joint Conference on Artificial Intelligence*, pp. 1275–1280.

Goodwin, C. 1981. Conversational Organization: Interaction between Speakers and Hearers. New York: Academic Press.

Gratch, J., A. Okhmatovskaia, F. Lamothe, S. Marsella, M. Morales, R.J. van der Werf, Louis-Philippe Morency. 2006. Virtual rapport. *Proceedings of International Virtual Agents*, pp. 14–27.

Jefferson, G. 1989. Preliminary notes on a possible metric which provides for a standard maximum silence of approximately one second in conversation. In Derek Roger and Peter Bull (eds.), Conversation: An Interdisciplinary Perspective, pp. 166–196.

Jonsdottir, G.R. 2008. A Distributed Dialogue Architecture with Learning. Master's thesis. Reykjavik University.

Jonsdottir, G.R., J. Gratch, E. Fast and K.R. Thórisson. 2007. Fluid semantic back-channel feedback in dialogue: Challenges and progress. In *IVA '07*, pp. 154–160. Springer.

Jonsdottir, G.R., K.R. Thórisson and N. Eric. 2008. Learning smooth, human-like turntaking in realtime dialogue. In IVA '08: Proceedings of the 8th international conference on Intelligent Virtual Agents, pp. 162–175. Berlin, Heidelberg: Springer-Verlag.

Moore, R. 2007. Presence: A human-inspired architecture for speech-based human-machine interaction. *IEEE Trans. Comput.*, **56(9):**1176–1188.

Ng-Thow-Hing, V., T. List, K.R. Thórisson, J. Lim and J. Wormer. 2007. Design and evaluation of communication middleware in a distributed humanoid robot architecture. In Prassler, E., Nilsson, K., Shakhimardanov, A. eds.), IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'07) Workshop on Measures and Procedures for the Evaluation of Robot Architectures and Middleware.

Nivel, E. and K.R. Thórisson. 2008. Prosodica Realtime Prosody Tracker. Technical Report. Reykjavik University Department of Computer Science. Technical Report RUTR-CS08001.

Pierrehumbert, J. and J. Hirschberg. 1990. The meaning of intonational contours in the interpretation of discourse. In Cohen, P.R., Morgan, J. and Pollack, M. (eds.), Intentions in Communication, pp. 271–311. Cambridge, MA: MIT Press.

Raux, A. and M. Eskenazi. 2007. A multi-layer architecture for semi-synchronous event-driven dialogue management. In *ASRU*, pp. 514–519, Kyoto, Japan.

Raux, A. and M. Eskenazi. 2008. Optimizing endpointing thresholds using dialogue features in a spoken dialogue system. In Proceedings of the 9th SIGdial Workshop on Discourse and Dialogue, pp. 1–10. Columbus, Ohio: Association for Computational Linguistics.

Sato, R., R. Higashinaka, M. Tamoto, M. Nakano and K. Aikawa. 2002. Learning decision trees to determine turn-taking by spoken dialogue systems. In ICSLP '02, pp. 861–864.

Schaffner, K.F. 2006. Reduction: The Cheshire cat problem and a return to roots. In *Synthese*, pp. 377–402. The Netherlands: Springer. Schlangen, D. 2006. From reaction to prediction: Experiments with computational models of turn-taking. In Proceedings of Interspeech 2006, Panel on Prosody of Dialogue Acts and Turn-Taking, Pittsburgh, USA.

Schwabacher, M. and A. Gelsey. 1996. Multi-level simulation and numerical optimization of complex engineering designs. In 6th AIAA/NASA/USAF Multidisciplinary Analysis & Optimization Symposium, AIAA-96-4021.

Stivers, T., N.J. Enfield, P. Brown, C. Englert, M. Hayashi, T. Heinemann, G. Hoymann, F. Rossano, J.P. de Ruiter, K.-E. Yoon and S.C. Levinson. 2009. Universals and cultural variation in turn taking in conversation. Proceedings of the National Academy of Sciences, **106(26):**10587–10592.

Sutton, R.S. and A.G. Barto. 1998. Reinforcement Learning: An Introduction. Cambridge, MA: The MIT Press, .

Thórisson, K.R. 1993. Dialogue control in social interface agents. In Inter- CHI Adjunct Proceedings, pp. 139–140. ACM Press, New York.

Thórisson, K.R. 1996. Communicative Humanoids: A Computational Model of Psycho-Social Dialogue Skills. Ph.D. thesis. Massachusetts Institute of Technology.

Thórisson, K.R. 2002a. Machine perception of multimodal natural dialogue. In McKevitt, P., Nulláin, S.Ó. and Mulvihill, C. (eds.), Language, Vision & Music, pp. 97–115. John Benjamins Publishing Co., Amsterdam, The Netherlands.

Thórisson, K.R. 2002b. Natural turn-taking needs no manual: Computational theory and model, from perception to action. In Granström, Björn, D. House, I. Karlsson (eds.), Multimodality in Language and Speech Systems, pp. 173–207. Dordrecht, The Netherlands: Kluwer Academic Publishers.

Thórisson, K.R. 2008. Modeling multimodal communication as a complex system. In Wachsmuth, I. and Knoblich, G. (eds.), Modeling Communication with Robots and Virtual Humans, pp. 143–168. Springer, Berlin, Germany.

Thórisson, K.R., H. Benko, A. Arnold, D. Abramov, S. Maskey and A. Vaseekaran. 2004. Constructionist design methodology for interactive intelligences. *A.I. Magazine*, **25(4):**77–90.

Thórisson, K.R. and G.R. Jonsdottir. 2008. A granular architecture for dynamic realtime dialogue. In Proceedings of the 8th International Conference on Intelligent Virtual Agents, pp. 131–138. Springer, Berlin, Germany.

Thórisson, K.R., G.R. Jonsdottir and E. Nivel. 2008. Methods for complex single-mind architecture designs. In Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS), pp. 1273–1276, Richland, S.C. International Foundation for Autonomous Agents and Multiagent Systems.

Thórisson, K.R., T. List, J. DiPirro and C. Pennock. 2004. OpenAIR: A Publish-Subscribe Message and Routing Specification 1.0. Technical Report.

Traum, D.R. and P.A. Heeman. 1996. Utterance units and grounding in spoken dialogue. In Proc. ICSLP '96, pp. 1884–1887, Philadelphia, PA.

Wilson, M. and T.P. Wilson. 2005. An oscillator model of the timing of turntaking. *Psychonomic Bulletin Review,* **38(12):**957–968.

Wood, M.F. and S.A. Deloach. 2000. An overview of the multiagent systems engineering methodology. In The First International Workshop on Agent-Oriented Software Engineering (AOSE-2000), pp. 207–221.

Wooldridge, M., N.R. Jennings and D. Kinny. 2000. The gaia methodology for agent-oriented analysis and design. *Autonomous Agents and Multi-Agent Systems*, **3(3):**285–312.